A Primer On The Standardization of Homomorphic Encryption

Enabling secure and scalable end-to-end data protection and privacy controls

A Duality Technologies White Paper

Introduction

It’s no secret: the power of data lies at the core of Digital Transformation. Thanks to improvements in data collection and storage, data science and analytics, and machine learning and AI, data has become a resource to be harnessed for extracting new insights and creating new forms of value. This revolution is transforming existing industries while creating new ones built upon the collection and leveraging of personal data.

However, as more and more data is collected, consumers and regulators are increasingly concerned about the impact on personal privacy. Revelations such as the Facebook-Cambridge Analytica scandal highlight how the business of data puts the personal privacy of consumers at risk. While industries such as financial services and healthcare have long managed data privacy regulations such as HIPAA, PCI, and the GLBA, landmark legislation such as the GDPR in Europe and the Consumer Privacy Act in California is formalizing the legal framework around protecting personal data to a broader range of enterprises and placing concepts such as “privacy-by-design” at the forefront of information systems.

Historically, there has been a tradeoff between privacy and utility when it comes to data. Given the reliance of machine learning (and therefore AI) and advanced analytics on access to data sets, it would seem that privacy and utility are fundamentally at odds—progress in one must come at the expense of the other. However, advancing both privacy and utility is not a zero-sum game.

Resolving this tension is an emerging class of technologies known as Privacy-Enhancing Technologies (PETs). PETs may refer to a wide-range of technologies that achieve specific privacy or data protection functionality, but an emerging subset of these technologies and methods allow for preserving privacy in machine learning and analytics. This growing field has been referred to as Privacy Preservation in Analytics by Gartner, while some refer to it as Privacy-Preserving Machine Learning.

PETs such as Homomorphic Encryption, Differential Privacy, and Secure Multi-Party Computation meet an urgent need to leverage data analytics and AI while protecting personally-identifiable information and remaining compliant with regulators. They also enable a growing desire to unlock the power of data-sharing between organizations. In regulated industries such as financial services and healthcare, privacy regulations have precluded organizations from collaborating on data to solve problems in novel ways, such as joint fraud and financial crimes analytics in the financial industry or large-scale multi-center medical or genomic studies using personal health data.

Privacy-enhancing technologies are the key to creating a sustainable data economy and unlocking new forms of value while protecting individual privacy. With the COVID-19 pandemic sweeping the globe, the need for technologies that allow the utilization of sensitive data to solve new global challenges while protecting privacy has taken on a new urgency.

These PETs differ in their applicability and maturity, but they all require the establishment of clear industry standards to ease implementation, ensure compliance with regulators, and facilitate mass adoption by consumers.

In this white paper, we take a look at these PETs, with a focus on homomorphic encryption, the need for standardization, and efforts underway in the homomorphic encryption community to create a lasting industry standard.

What is Homomorphic Encryption?

Data has three basic states: at-rest, in-transit, and in-use. Organizations that handle sensitive data such as personally-identifiable information (PII) typically encrypt or otherwise protect that data while it is being stored (at-rest) or when it is being transmitted (in-transit). However, whenever the data needs to be processed, analyzed, or manipulated in any way, it must first be decrypted - leaving it vulnerable.

Fully Homomorphic Encryption (FHE), commonly also referred to as Homomorphic Encryption, finally solves this problem. Homomorphic Encryption is a set of encryption methods that allows computations to be performed on encrypted data without ever decrypting it, keeping data secure in all three states.

Homomorphic encryption schemes use a mapping between two algebraic structures that preserves the operations of the structures. This allows analytical functions to be run directly on encrypted data (ciphertexts), while yielding the same encrypted results as if the functions were run on plaintext.

The Holy Grail of Data Privacy

Because it secures data from end-to-end in all three states, homomorphic encryption has long been dubbed the “Holy Grail of data privacy” or the “Holy Grail of cryptography.”

The idea of homomorphic encryption is not new. Cryptographers have discussed the idea since it was proposed in 1978, but didn’t know if it was possible. It wasn’t until 2009 when Craig Gentry, then at Stanford, described the first plausible construction for a fully homomorphic encryption scheme, showing that FHE could be realized in principle.

The Glove Box

Homomorphic encryption was described by Gentry with a “glove box” analogy. Imagine the owner of a jewelry store wants her employees to assemble precious materials into finished jewelry, but she is worried about theft. She addresses the problem by constructing glove boxes for which only she has the key. Anyone can stick their hands inside the gloves and manipulate what’s inside the box, but they can’t take the materials (raw data) out from the box. When they finish working with the raw materials to produce the finished product, only the person who has the key (private cryptographic key) can open the box to remove the now-assembled product (processed data).

New Opportunities for Collaboration on Data

FHE is arguably the most important breakthrough in theoretical computer science of the 21st century. Since Gentry's breakthrough, research and implementation efforts throughout academia, government, and industry have brought FHE from theory to reality.

HE enables computations, including machine learning and AI analysis, on encrypted data, allowing data scientists, researchers, and enterprises to gain valuable insights without decrypting or exposing the underlying data or models – allowing organizations to extract value from data while maintaining privacy and complying with data privacy regulations. This provides a functional privacy, eliminating the tradeoff between data and utility.

This is particularly useful where enabling collaborations between parties with sensitive data – such as privacy-preserving collaborations with patient data between multiple healthcare and research centers, or inter-bank cooperation in financial crime investigations. Joint analyses on merged data sets can be run without ever exposing the data to either party.

Because homomorphically-encrypted data is encrypted from end-to-end in all three states, no trusted third parties are ever required. This allows for computations to be outsourced, keeping both the data and the analytical models used to operate on the data safe, secured, and unrevealed. A cloud host could run a computation on the data, get an encrypted result, and give that result back to the data owner. They could then decrypt that result, with the decrypted result being the same as if they had run the computation on the original data without encryption.

Example Applications of Homomorphic Encryption

Data-sharing and analytics in regulated industries. In highly-regulated industries such as health care, education, or financial services, organizations can share and run joint analytics on user data without ever exposing sensitive personal data.

Secure cloud storage and processing. Highly-regulated industries can securely outsource data storage and processing to cloud environments, while retaining the ability to calculate and search ciphered information. Multiple parties can securely share private data with a third party to analyze, and have the results returned to one or more of the participants to be decrypted.

Privacy in consumer Internet services. Internet services could be provided to users without exposing their data to the service providers or third-parties. If used by large privacy-sensitive data aggregators such as Facebook, homomorphic encryption would allow consumers to protect their information while allowing companies to still derive valuable insights from the data.

Sidebar: What is “Fully” Homomorphic Encryption?

Homomorphic encryption schemes can be grouped into three main types: Partially Homomorphic Encryption (PHE), Somewhat Homomorphic Encryption (SHE) and Fully Homomorphic Encryption (although you may see other smaller distinctions made as well). The primary difference between these three has to do with the types and frequency of mathematical operations that can be performed on their ciphertext.

PHE schemes only allow only select mathematical functions to be performed on encrypted data, while SHE schemes support limited operations but that can only be performed a set number of times. Partially homomorphic encryption is the foundation for RSA encryption.

What we are discussing in this whitepaper is Fully Homomorphic Encryption. FHE is the gold standard of homomorphic encryption that allows for the use cases we discuss here to keep data secure and accessible. It wasn’t known if a Fully Homomorphic Encryption scheme was possible until Craig Gentry proposed a plausible FHE scheme using lattice-based cryptography.

How Does Homomorphic Encryption Compare to Other PETs?

Previous approaches to protect privacy relied mostly on de-identification and anonymization techniques, which typically involves removing PII fields from data sets. However, recent studies *(add footnotes to Nature study and Scientific American article here)* have demonstrated that advancements in machine learning capabilities enable re-identification of anonymized data - rendering anonymization technologies insufficient as a means of protecting privacy.

PETs such as homomorphic encryption, multi-party computing (MPC), and differential privacy are introducing new paradigms for data protection with strong mathematical guarantees. These technologies and methodologies each offer different approaches that can be used together in a complimentary fashion, or individually when they are best suited to particular use cases.

Differential Privacy

Differential privacy is a system for protecting sensitive data by adding noise to the data, in order to make it impossible to reverse-engineer individual inputs. Differential privacy itself is not a technology, but a mathematical definition of privacy that measures a variety of techniques which increase the anonymity of data through the addition of noise.

Differential privacy is useful for distinguishing trends and gaining broad insights from a large data set. For example, census data is often anonymized with noise to protect the privacy of individual respondents, and differential privacy will be used in the 2020 United States Census. However, differential privacy techniques are not a viable option for analyses that require precision, as the added noise makes the results of the analysis directionally correct, but not actually correct.

Using a third party to perform analytics with differential privacy techniques requires trust and presents a window of vulnerability. The party performing the calculation must see the data to determine the addition of noise. When combining data sets with other parties, it requires trust that the other parties are providing accurate data to the third party before the computation is performed.

Secure Multiparty Computation

Secure multiparty computation (SMPC) is a set of cryptographic protocols that distribute a computation across multiple parties, so no individual party can see the other parties' data and nobody sees the complete set of inputs.

While secure multiparty computation protocols can be a useful way to compute on distributed data in some cases, these methods are still relatively new and immature, making application in many cases limited. Data is not encrypted end-to-end, and since all parties know the output, they can use this to infer sensitive data.

SMPC systems are also expensive. They require a completely custom set-up for each use case, making set-up very expensive, and have high communication costs, making them costly to operate on an ongoing basis.

Secure Hardware Enclaves

In addition to mathematical and protocol-oriented approaches, hardware-based technologies such as secure enclaves provide a different approach by enforcing security at the hardware level by the CPU itself. Secure enclaves, such as those created by Intel SGX, are isolated memory locations which can be used to protect applications and data in-use while in a trusted execution environment. Hardware-level encryption isolates the software from the hardware and OS, making it very difficult for attackers to access and decrypt private data, even with access to the hardware. While secure enclaves can be best for some use cases, there are limitations on both the size of computations and collaboration ability when compared to other approaches.

Homomorphic Encryption

Homomorphic encryption is the only solution to keep data end-to-end encrypted and secure in all three states. Using HE provides ongoing protection in a way a solution like Differential Privacy does not. For example, even if a machine stores your data without authorization, the data remains protected by encryption.

Homomorphic encryption is akin to "the Swiss Army knife" of privacy technologies. It can be used with any hardware or software setup, and doesn't require the cooperation of other parties. Since the results of computations on homomorphically-encrypted data are the same as if they were on plaintext, HE is the best option for accurate individual-level insights and exact results.

The downside to homomorphic encryption is that techniques and standards are still evolving, so some techniques are not fully homomorphically-encrypted and computations can be slow and bandwidth-heavy. Current HE implementations are also not very developer-friendly and can be difficult to use. HE is on the path towards standardization, but as of now, there is still work to be done.

Comparing PETS

In the chart below, we compare these PETS on the following factors: hardware dependence, end-to-end encryption, exactness of general computation results, accuracy for individual-level insights, support for cryptographic access delegation, existing industry standards, and ability to collaborate on multiple data sets.

(Insert comparison charts here)

How Does Homomorphic Encryption Compare to Classical Encryption?

A Short History of Encryption

The use of cryptography dates back thousands of years, with evidence of some use of encryption (the encoding of information) found in most early civilizations. Symbol replacement, an early form of encryption, appears in Egyptian and Mesopotamian writings, with the earliest evidence found on the tomb of Egyptian noble Khnumhotep II, who lived approximately 3,900 years ago.

Cryptographic techniques such as substitution ciphers were widely used in Ancient Greece and Rome and throughout the Middle Ages to protect military information. The 20th century saw the introduction of mechanical encryption/decryption devices - most famously, the German Enigma machine - the cracking of which became a major Allied achievement in World War II.

The post-war era and the advent of the computer led to the birth of modern cryptography, but two events in the 1970s brought encryption into wide commercial use.

The Adoption of an Encryption Standard

The first was the adoption of the first encryption standard in the US in 1973, known as the *Data Encryption Standard (DES)*, by the National Bureau of Standards (now called NIST). The DES was based on a block cipher a research group at IBM developed to protect customer data.

DES was the first attempt to create a universal encryption standard and is one of the most widely-used crypto systems of all time. However, as computing power increased by the late 90s, the DES had been cracked, leading to NIST putting out a request for a new block cipher. After an open competition, NIST selected an entry from two Belgian cryptographers named Rijndael, and christened it the *Advanced Encryption Standard (AES)*.

With the publication of FIPS 197 in 2001, the AES officially replaced the DES. Today, the AES is available royalty-free worldwide and is the standard for use by the United States government.

The Invention of Public-Key Cryptography

The second major event was the invention of public-key cryptography. AES and DES are what are known as symmetric-key or private-key encryption, which means the same key is used for both encryption and decryption. In a public-key system, different keys are used for each function - the encryption key is made public and available for anyone to see, while a private key is needed to decrypt a message.

Public-key cryptography is used for implementing digital signature schemes, such as the RSA and DSA crypto systems, which allow for the signing and verification of digital messages. Public-key algorithms underpin widely-used Internet protocols and network security schemes such as PGP, Transport Layer Security (TLS) and its predecessor SSL, the EMV payment standard, SSH, and Bitcoin. These schemes have become an essential component of securing data at-rest and in-transit across the Internet, from emails and messaging, to visiting websites, to eCommerce transactions and transmitting financial information.

Quantum Computing and The Future of Encryption

As computing power advances, so must the quality of encryption. Encryption schemes are never perfect, and as computing hardware gets faster and mathematical analysis of encryption methods advances, new approaches are discovered to penetrate encrypted data.

Encryption effectively works by showing the answer to a question and only letting someone in if they know what the right question is. Attempts to hack encrypted services are thwarted by the use of long, complex prime numbers which require too high a computational workload to guess and so can only be determined by the use of cryptography keys.

The computational complexity of difficult problems from number theory form the security basis for public-key algorithms. These problems are known as computational hardness assumptions, which is a hypothesis that a particular problem cannot be solved efficiently. Different crypto systems use different hardness assumptions - the hardness of RSA is related to the integer factorization problem, Diffie–Hellman and DSA are related to the discrete logarithm problem, and the security of elliptic curve cryptography is based on number theoretic problems involving elliptic curves.

The development of quantum computing could lead to current encryption methods becoming obsolete, as most popular public-key algorithms have been discovered to be vulnerable to attacks from quantum computers. The integer factorization problem, discrete logarithm problem, and the elliptic-curve discrete logarithm problems mentioned above can all be solved on a sufficiently powerful quantum computer running an algorithm known as Shor's algorithm.

While the practical development and adoption of quantum computing may still take two decades, cryptographers are working on developing new encryption methods that will be secure against quantum computers. In 2016, NIST initiated a process to solicit and standardize post-quantum cryptographic algorithms.

Post-Quantum Homomorphic Encryption Schemes

Several of the submissions to NIST's post-quantum cryptography standardization project use lattice-based cryptographic constructions, which are based on the assumption that certain well-studied computational lattice problems cannot be solved efficiently. Lattice-based cryptography is the leading candidate for public-key, post-quantum cryptography, and are currently considered to be secure against both classical and quantum computers.

Practical homomorphic encryption schemes are based on what is known as the *Ring-Learning with Errors (RLwE)* problem, a hard mathematical problem based on high-dimensional lattices similar to constructions submitted to NIST. A long history of peer-reviewed research confirming the hardness of the RLWE problem indicates that they are at least as secure as any currently-standardized encryption scheme while also being secure against quantum computers.

How Does Homomorphic Encryption Relate to Classical Encryption Schemes?

Like other forms of public-key cryptography, homomorphic cryptosystems use a public key to encrypt data, which allows only the individual with the matching private key to access the unencrypted data. In this way, homomorphic encryption is similar to RSA and the public-key crypto systems mentioned earlier, and has *public-key properties*.

However, where it differs is that it uses an algebraic system that allows for computations (operations) to be performed on the encrypted data. This is what makes homomorphic encryption more useful than classical encryption systems, which can only encrypt data in-transit or at-rest, but not in-use.

Practical homomorphic encryption schemes provide efficient instantiations of post-quantum, public-key encryption schemes, making them the leading option for a future end-to-end cryptographic standard.

Can Homomorphic Encryption Be Trusted?

Throughout the past decade, the development of HE has been driven by a broad collaborative community spanning academia, government, and industry. In academia, homomorphic encryption schemes have been developed worldwide by leading researchers in lattice-based cryptography and subjected to rigorous peer-review in academic journals.

Transforming that research into implementation has been a broad user and contributor community driving open-source development of libraries implementing practical HE schemes.

There are at least 6 research groups around the world who have made libraries for general-purpose homomorphic encryption available: SEAL, HElib, PALISADE, cuHE, NFLLib, and HEAAN. All of these are based on RLWE-based systems and implement one of three widely-tested encryption schemes. A number of other small open-source libraries primarily used by academic researchers have also been created.

Leading implementations of HE schemes have open security settings that have been widely-tested and vetted by a robust global community of privacy professionals, corporate research teams, defense contractors, and academic researchers.

Academic Background

Homomorphic encryption as a concept goes back over 40 years. The problem of constructing a fully homomorphic encryption scheme was proposed by Ron Rivest, Leonard Adleman, and Michael Dertuozos in “On data banks and privacy homomorphisms” in 1978. A year earlier, Rivest and Adleman had devised the popular RSA cryptosystem (which itself is partially-homomorphic) with Adi Shamir. For the next 30 years, cryptographers proposed partially-homomorphic schemes such as the ElGamal and Goldwasser-Micali cryptosystems, but it was unclear if a fully homomorphic scheme was possible.

In 2009, a breakthrough was made when Stanford doctoral student Craig Gentry, using lattice-based cryptography, described the first plausible construction for a fully homomorphic encryption scheme. Gentry had created an *existence proof*, showing that a fully homomorphic encryption scheme was possible, but he didn’t have a real-world implementation of the concept. This development is arguably the first and most important breakthrough in theoretical computer science of the 21st century.

Early Implementation Efforts and US Government Support

An early implementation of homomorphic encryption was developed by IBM in 2010 which showed how two bits could be encrypted and then a Boolean AND operation could be run on this operation in a half hour. Although not close to practical at that time, these early results garnered great interest in the government and academic communities, sparking a wave of new theoretical advancements, implementations, and testing.

DARPA, the premier blue-sky research funding agency of the US military, developed a large research funding effort to support the development of homomorphic encryption. This initial program, called PROCEED, was created to develop secure cloud computing technologies.

During the course of the PROCEED program, a team led by Dr. Kurt Rohloff (then senior scientist at Raytheon BBN Technologies) developed implementation techniques which improved the runtime performance (and practicality) of homomorphic encryption by more than 6 orders of magnitude, while building an HE library with modular open design principles that support rapid application development with multiple FHE schemes.

PALISADE Open-Source HE Library

In 2014, Dr. Rohloff left his position at Raytheon to continue DoD-funded efforts in collaboration with cryptography pioneers and MIT professors Shafi Goldwasser and Vinod Vaikuntanathan, whose efforts resulted in the PALISADE open-source library. Based on insights from two earlier generations of HE implementations and utilizing the modular design used in the PROCEED program, PALISADE was designed with a focus on practical applications of homomorphic encryption.

PALISADE development has been driven on a project basis with initial funding from the DARPA PROCEED and SafeWare programs, with subsequent improvements funded by the DARPA MARSHAL program, IARPA (RAMPARTS initiative), the National Security Agency, National Institute of Health, the United States Navy and Office of Naval Research, the Sloan Foundation, and the Simons Foundation. PALISADE has also relied on a strong user and contributor community from a wide range of academic institutions and with industry support from Raytheon (BBN), IBM Research, EuroCom, CACI/Lucent, Vencore Labs, Galois, and Two Six Labs.

Corporate Involvement

In addition to government backing and academic work, major support and library development from teams at IBM, Microsoft, and Intel has greatly furthered practical implementations of fully homomorphic encryption.

IBM: HElib Open-Source HE Library

IBM played a key role in getting early HE implementations off the ground, and in 2013 released HElib, a software library for HE researchers that implements the BGV scheme and was intended as a sort of "assembly language for HE.”

Microsoft: SEAL Open-Source HE Library

In 2015, Microsoft Research released the first version of a homomorphic encryption library called Microsoft SEAL (Simple Encrypted Arithmetic Library) with the goal of providing a well-engineered and documented homomorphic encryption library, free of external dependencies, that would be easy for both cryptography experts and novice practitioners to use. In 2018, Microsoft opened SEAL up to open-source use.

Intel: HE-Transformer Open-Source Tool

In 2018, Intel released an open-source tool called the HE-Transformer, a HE backend to nGraph, Intel’s neural network compiler. Using the Microsoft SEAL library to implement the cryptographic schemes, HE-Transformer enables data scientists to develop neural networks on popular open-source frameworks such as TensorFlow then deploy them to operate on encrypted data.

The Importance of Standardization

Standardization is so integral to our lives that we don't even notice it. Imagine the frustration you would have if you bought a light bulb and it couldn't fit it into your lamp, or your lamp couldn't plug into the socket. Standardization ensures that new technologies work seamlessly, integrate into our current systems, and establish trust so that markets can operate smoothly.

The International Organization for Standardization (ISO), the world's leading standards body, defines a standard as "a document that provides requirements, specifications, guidelines or characteristics that can be used consistently to ensure that materials, products, processes and services are fit for their purpose." Standards make interoperability possible, protect consumers by ensuring safety and quality, and provide a common language to measure and evaluate performance.

Creating Standards: Standards Bodies and Voluntary Consensus Organizations

Standards are typically defined by national standards bodies (such as the US Department of Commerce's NIST) and international standards bodies (which often focus on a particular industry or type of technology) such as ISO, the International Telecommunications Union (ITU), and the Institute of Electrical and Electronics Engineers (IEEE). Standardization arises out of voluntary cooperation among industry, consumer groups, governments, researchers, and other interested parties who develop technical specifications based on consensus.

To keep up with the breakneck pace of technological development, technical standards are increasingly created by industry consortia or other voluntary consensus organizations, sometimes later becoming adopted by standards bodies when they have reached a mature level of acceptance by industry.

For a simple example of the importance of voluntary standards bodies, consider the impact the World Wide Web Consortium (W3C) and the Internet Engineering Task Force (IETF) have had on standardizing the Internet and the Web for mainstream use on a global scale. Without the development and universal acceptance of the W3C recommendations such as HTML, HTTP, or URL, CSS, or the IETF's TCP/IP, there would be no functioning global internet as we know it.

Cryptographic Standards

Standards such as the DES (and later AES) have played a major role in making encryption accessible to the private sector. Cryptographic standards are published by various bodies including NIST, the IETF, and more. Perhaps the best known are the United States government's Federal Information Processing Standards (FIPS) publications, which determine standards for use by federal government agencies. FIPS publications coexist with industry standards created by voluntary standards organizations, and NIST works with these organizations to harmonize standards.

The NSA's Suite B (now replaced by the CNSA suite) of algorithms provide guidance on strong cryptographic algorithms and secure protocol standards, recommending widely-used standard crypto systems such as RSA, AES, and elliptic-curve protocols. However, the NSA is currently preparing a transition to a forthcoming suite of quantum-resistant algorithms.

Standardized protocols such as those specified in FIPS publications help guide organizations handling sensitive data on creating their own lists of approved crypto systems to protect data integrity, secure communications, and minimize risk.

Why PETS Need Standardization

The urgent need for widespread adoption of PETs is clear. With data privacy regulations increasing worldwide, companies need a way to collaborate on data and maximize data utility while protecting privacy and staying compliant.

However, current solutions are unproven, have unclear impacts on compliance, and are often difficult to implement. Like with existing widely-deployed crypto systems, clear and robust standards are critical to winning consumer trust and confidence,

Security. Organizations need to have confidence that the security assumptions and implementation schemes underlying emerging PETs have been thoroughly and rigorously documented, reviewed, and tested, and best practices have been defined that represent the latest collective knowledge.

Implementation. Standards disseminate the knowledge and practices, such as a set of agreed upon parameters for implementing algorithms and schemes, and help create an ecosystem of developer tools to ease implementation and connect them to current business systems. A mature and interoperable ecosystem ensures investments made in PETs will

Legal and Compliance. Standardization is key to legal on several fronts. Most importantly, standards help ensure organizations that the adoption of a given technology is not increasing their potential for regulatory liability. Standardization also removes the complexity of drafting legal agreements - which can be very time consuming when done on a custom basis. It streamlines the process for contracting and removes a major barrier to collaboration between two parties, as contracts can reference existing standards as guidelines for implementation.

Sidebar: Homomorphic Encryption and Regulatory Compliance

There is a wide range of data privacy regulations for industries that handle sensitive personal data, including the California Consumer Privacy Act, the GDPR and European Banking

Authority regulation in the EU, GLBA and SOX Act regulations for the financial industry and HIPAA for healthcare in the United States, and more.

While they differ in their specific prescriptions and guidelines, most compliance regulations focus on the protection of sensitive data at rest, during transactions, and in transit through network connections. Some regulations require particular technologies, but encryption is widely considered the most effective means of protection and is effective in satisfying all major data privacy regulations.

Standardized FHE schemes are well-tested and secure forms of end-to-end encryption that are considered to be at least as secure as existing encryption standards. In the near future, it is expected regulators will begin to incorporate guidelines around usage of PETs for secure data sharing and processing.

Homomorphic Encryption Standardization

In 2015, the leading open-source FHE libraries (PALISADE, SEAL, and HElib) came together to try to standardize protocols and security parameters for their users. This early community of teams from Microsoft, IBM, Intel, and the PALISADE community engaged NIST and ISO to develop an HE standard, and they encouraged the group to create an industry consortium to draft standards.

This formed the Homomorphic Encryption Standards consortium (homomorphicencryption.org), which held its first standardization workshop in 2017. Since then, standards meetings featuring 50-100 participants have been held every 6 months, further developing the standard and disseminating knowledge on HE.

Participants have included major technology firms such as IBM, Microsoft, Intel, SAP, Intuit, and Google, financial institutions such as MasterCard and major banks, academic institutions such as MIT, UCSD, and USC, and relevant government agencies such as NIST, the NSA, the NIH, the US Navy, and the Canadian CSE.

The first standardization workshop led to the creation of three white papers addressing the *Security*, *API*, and *Applications* of homomorphic encryption, and serve as the basis for the development of the *Homomorphic Encryption Standard.*

Standardizing Security

The security properties of RLWE-based homomorphic encryption schemes can be hard to understand, so the standard presents the security properties of standardized schemes in a clear and understandable form.

The security white paper outlines current collective knowledge regarding the security of standardized FHE schemes, specifies the schemes, and recommends a wide selection of parameters to be used for homomorphic encryption at various security levels. It describes the security assumptions of the LwE and RwLE problems, whose hardness form the basis of the HE schemes; reviews known attacks and uses them to suggest concrete parameter choices; and describes additional features of the schemes which make them useful in different applications and scenarios.

The security standard focuses on the two primary schemes for HE implementation: the Brakerski-Gentry-Vaikuntanathan (*BGV*) scheme and the Brakerski/Fan-Vercauteren (*BFV)* scheme. The BGV and BFV schemes have been widely-used by the HE community. The standard also describes the *GSW* scheme, which shows promise for future development, while an upcoming standard is expected to include the *CKKS* scheme.

The Homomorphic Encryption Standard

Security has been the first priority in HE standardization. After a public comment period and community review, the security white paper was publicly endorsed by leading security experts at the second standardization workshop. resulting in the first version of the *Homomorphic Encryption Standard*. The co-authors of the standard include the leading members of the lattice cryptography and cryptanalysis community from throughout academic and industry.

Efforts to create standardized APIs to ensure library interoperability and create developer tools and an SDK for a higher-level HE ecosystem are underway. The *API* white paper lays the foundation for an API standard by describing a storage model and programming model to make it easier for application developers to adopt and integrate HE implementations. In a standardized HE ecosystem, applications will be portable, offering users a maximum of flexibility and business value.

Conclusion

Thanks to a strong and active community spanning government, academia, and industry, lattice-based, public-key, fully homomorphic encryption has been transformed from concept to mature technology ready for widespread adoption.

Now, standardization is opening the market to a broad range of participants and gaining traction with the world's foremost international standards bodies. Since its first publication in 2017, the *Homomorphic Encryption Standard* has gained the attention of NIST, the IETF, ISO, and the ITU, and is rapidly moving towards recognition as a mature and accepted standard for the secure implementation of this promising PET.

As the dialogue about privacy rights continues in the media, legislative bodies, and courtrooms around the globe, one thing is clear: the future of data will put user privacy and consent into the driver's seat. But thanks to PETs such as homomorphic encryption, that advancement does not need to come at the price of utility.

This paradigm shift will enable a sustainable and privacy-protecting data economy, but it will require standardized, trustworthy, secure, mature, interoperable, and efficiently implementable mechanisms at its core.