**Efficient Homomorphic Encryption for Private Cloud Computing: A Comparative Study of BFV and CKKS Schemes.**

**Research Objectives:**

1. To evaluate the performance of the BFV and CKKS homomorphic encryption schemes in terms of computation time, memory usage, and communication overhead.
2. To compare the security guarantees provided by the BFV and CKKS schemes, and analyse the trade-off between security and efficiency.
3. To demonstrate the practicality of the BFV and CKKS schemes in a private cloud computing environment by implementing and evaluating their performance on a cloud platform.

Methodology: The proposed research will use a comparative study methodology to evaluate the performance and security of the BFV and CKKS homomorphic encryption schemes. The study will be conducted in three phases:

1. ***Implementation:*** The BFV and CKKS schemes will be implemented in a private cloud computing environment using appropriate libraries and tools. The implementation will be evaluated in terms of computation time, memory usage, and communication overhead.
2. ***Performance evaluation***: The performance of the BFV and CKKS schemes will be evaluated using a benchmark suite that includes various homomorphic operations, such as addition, multiplication, and re-linearization. The evaluation will be conducted on a range of input sizes and security parameters.
3. ***Security analysis***: The security of the BFV and CKKS schemes will be analysed by evaluating their resistance to known attacks, such as side-channel attacks and ciphertext-only attacks. The analysis will also consider the impact of parameter choices on the security of the schemes.

**Problem statement**

The increasing demand for cloud computing has led to an increased need for secure and private cloud computing solutions. Homomorphic encryption is a promising solution for enabling secure computations on encrypted data without the need for decryption. However, existing homomorphic encryption schemes may not be efficient enough for practical use in cloud computing applications.

Efficient Homomorphic Encryption for Private Cloud Computing is a problem statement that aims to address the following issues:

1. ***Performance***: Existing homomorphic encryption schemes may not be efficient enough for practical use in cloud computing applications, especially for large datasets. The computational and memory requirements of homomorphic encryption schemes can be significant, which can lead to slow and inefficient computations.
2. ***Security***: While homomorphic encryption schemes offer strong security guarantees, they are vulnerable to side-channel attacks, which can compromise the security of the encrypted data.
3. ***Practicality***: Homomorphic encryption schemes require specialized knowledge and skills to implement and use effectively, which can limit their practicality for cloud computing applications.

**Conceptual Framework**

In the context of Efficient Homomorphic Encryption for Private Cloud Computing, the dependent variable is typically the efficiency of homomorphic encryption. This can be measured in terms of factors such as computation time, memory usage, network bandwidth, and power consumption. The efficiency of homomorphic encryption can be affected by a range of independent variables, including:

1. Encryption algorithm: The choice of encryption algorithm can impact the efficiency of homomorphic encryption.
2. Key size: The size of the encryption key can impact the efficiency of homomorphic encryption.
3. Cloud infrastructure: The performance of homomorphic encryption can be affected by the cloud infrastructure on which it is executed.
4. Data size: The size of the data being processed can impact the efficiency of homomorphic encryption.
5. Homomorphic operations: The specific homomorphic operations being performed can impact the efficiency of homomorphic encryption.
6. Application requirements: The requirements of the application being developed can impact the efficiency of homomorphic encryption.
7. Security level: The level of security required for the application can impact the efficiency of homomorphic encryption.

Homomorphic Encryption: This component involves the use of mathematical techniques to encrypt data in such a way that computations can be performed on the ciphertext without revealing the underlying plaintext. This enables data privacy to be preserved while still allowing computation on the data in a private cloud computing environment.

Efficiency: This component involves the development of efficient homomorphic encryption algorithms and protocols that can be executed on resource-constrained devices such as mobile phones and Internet of Things (IoT) devices. The goal is to minimize the computational and communication overhead of homomorphic encryption, making it practical for real-world applications.

-Speed and efficiency of the processors

-Available memory

-Complexity of the data being encrypted

-Performance of other underlying hardware and software systems used for encryption, decryption and computation

**(Intervening Variable)**

-Encryption Algorithm

-Key size

-Cloud Infrastructure

-Data Size

-Security Levels

**(Independent Variable)**

Efficiency of Homomorphic Encryption.

-Computation time

-Memory usage

-Network Bandwidth

**(Dependent Variable)**

Security: This component involves ensuring the security of homomorphic encryption in private cloud computing environments. This includes protecting against attacks such as side-channel attacks, chosen ciphertext attacks, and replay attacks, among others. It also involves ensuring the integrity and confidentiality of the encrypted data.

Cloud Infrastructure: This component involves the development of cloud infrastructure that supports Efficient Homomorphic Encryption. This includes the deployment of cloud servers that can perform homomorphic computations on encrypted data, as well as the development of software frameworks that enable developers to easily integrate homomorphic encryption into their applications.

Applications: This component involves the development of applications that use Efficient Homomorphic Encryption in private cloud computing environments. Examples include secure data analytics, secure machine learning, and secure search, among others.

**Hypothesis**

By improving the efficiency of homomorphic encryption schemes, it is possible to enable secure and practical cloud computing applications that can process large amounts of sensitive data without the need for decryption.

This hypothesis is based on the assumption that improving the efficiency of homomorphic encryption schemes can lead to practical and secure cloud computing applications. By reducing the computational and memory requirements of homomorphic encryption schemes, it is possible to improve their efficiency and enable the processing of large datasets on cloud computing platforms without sacrificing security. This can be achieved through the optimization of existing homomorphic encryption schemes, or by developing new techniques that can improve their efficiency while maintaining their security guarantees.

Furthermore, this hypothesis assumes that the use of homomorphic encryption in private cloud computing applications can provide a high level of privacy and security, which is essential for handling sensitive data. The hypothesis also assumes that the improved efficiency of homomorphic encryption schemes can lead to practical applications that can be used by non-experts, thus increasing the accessibility of secure cloud computing solutions.

Overall, this hypothesis suggests that improving the efficiency of homomorphic encryption schemes can enable secure and practical cloud computing applications that can meet the growing demand for secure and private cloud computing solutions