CYBERNETICA

Sharemind HI White Paper

Document ID: D-16-190

Version 1.1
2023-01-12

1. Introduction

1.1. Purpose

Sharemind HI is a platform for the analysis of confidential data from multiple stakeholders. Sharemind HI provides full control over who can access the data and results, and provides cryptographic methods for enforcing such control. The analysis of confidential data runs on a central server which eases deployment. The Sharemind HI SDK makes developing secure data-driven applications easier by abstracting away key management and cryptographic measures.

Sharemind HI was created to reduce the risk of a privacy breach when processing confidential data. The data is encrypted at the source, by the Data Producer, and only then sent to the Sharemind HI service. Even the server administrator of the computer which hosts the service has neither access to the unencrypted data nor to the encryption keys. Sharemind HI does not remove the cryptographic protection from data even while processing it.
The data remains protected throughout the whole analysis.

1.2. Users

Sharemind HI is designed for the providers of data-driven services who need to provide better protection for the data they are analysing. Service providers can give strong assurances to data producers, analysts or third parties about the data only being used as agreed upon.

1.3. Technology

To provide the security guarantees, Sharemind HI relies on two technologies - cryptographic algorithms and a Trusted Execution Environment (TEE).

Cryptography is used to protect the data in transit and at rest, especially for long-term protection. The hardware based TEE helps to isolate the security sensitive parts of an application from the rest of the system.

The TEE technology used in Sharemind HI to implement the privacy-preserving data processing is Intel® Software Guard Extensions (SGX) which is available in modern Intel® processors.

2. Intel® Software Guard Extensions

Confidential data at rest and in transit can be protected with commonly available industry grade encryption. However, in traditional applications confidential data in use—encryption keys, business secrets—is not encrypted. While the operating system can offer some level of protection, nothing protects from the components (hypervisor, OS) or participants (the administrator) being malicious or compromised. Intel® Software Guard Extensions (SGX) is a technology to provide an extra layer of protection for the confidential data in use. To protect data in use, Intel® SGX applies these three key concepts: enclaves, attestation and data sealing.

2.1. Enclaves

Intel® SGX provides a set of CPU instructions for creating and running parts of an applications in an isolated environment called "enclave". Through Intel® SGX, the enclave's working memory is encrypted and the integrity of the enclave's working memory is protected. The encryption and integrity protection will shield the data in the memory even in the presence of a privileged malware such as a compromised operating system or hypervisor. The attack surface of an application can be significantly reduced by using enclaves.

Intel® SGX applications are separated into a trusted and an untrusted part. The smaller trusted part, which takes care of handling private information, is put in the enclave. The untrusted part is responsible for orchestrating the application as a whole. It creates the enclaves, reads and writes files, communicates over the network and processes data which is not privacy sensitive.

2.2. Attestation

Each enclave has attributes and measurements (a fingerprint of the compiled source code) which form the identity of an enclave. Attestation is a mechanism to cryptographically determine the identity of an active enclave. At the same time, attestation is used to establish a secure communication channel with the enclave. Intel® SGX offers two varieties of attestation: local attestation and remote attestation.

Local attestation is performed between two enclaves on the same CPU. Each enclave verifies that its peer is the expected one. Local attestation is essential for building applications where the trusted part is split into multiple enclaves which need to communicate with each other.

In remote attestation, a user verifies that a specific enclave was created within Intel® SGX on a remote machine. As a result, the user can trust that the enclave on the remote machine runs the agreed upon analysis and it is running on genuine hardware. The user can then upload their confidential data over the newly created secure communication channel for analysis.

An additional third party, for example the Intel® Attestation Service (IAS), is required for remote attestation which helps the user to verify the trustworthiness of the enclave.

2.3. Data Sealing

Data sealing enables enclaves to store data outside of the enclave's working memory without compromising the confidentiality and integrity of the data. Sealing the data means encrypting the data with a key that can only be derived by the same enclave on the same CPU. Data sealing is used to persist confidential data across system reboots, and to offload large volumes of data out of the enclave's working memory.

3. Sharemind HI Platform and Applications

3.1. Concepts

Sharemind HI is a client-server platform. The server-side application of a Sharemind HI solution consists of the following components:

- the solution agnostic Sharemind HI Server which itself consists of an untrusted application and management enclaves,
- **solution specific** enclaves (Task Enclaves) which contain the compiled source code of the algorithm that processes the data (Analysis Code), and
- a **solution specific** access control configuration.

The client-side application consists of the solution agnostic Sharemind HI Client library and **solution specific** additions. It audits the server-side enclaves, uploads data, invokes task enclaves and, when processing finished, downloads data. Security checks and encryption such as remote attestation, secure communication and data encryption, are either transparently performed by the Sharemind HI Client library or require only minimal intervention by the user.

The data encryption model of Sharemind HI is illustrated below. The input data, shown in blue, is encrypted at the client side and sent to the server. The encryption keys of the input data are securely transferred to the management enclaves. Task enclaves can decrypt and process the input data. Likewise, the output data, shown in green, is produced and encrypted inside of the task enclaves and stored on the server. When requested, the management enclaves securely transfer the output data encryption keys to clients.

At any point during the life cycle of a Sharemind HI Server, a client can request a cryptographic proof of the task enclaves in the server, shown in yellow on the figure. An auditor can generate the same cryptographic proof directly from the analysis code which they validated to be trustworthy or not. A client can thus compare the proof from the Sharemind HI Server with the proof from the auditor and conclude whether the task enclaves are trustworthy or not.

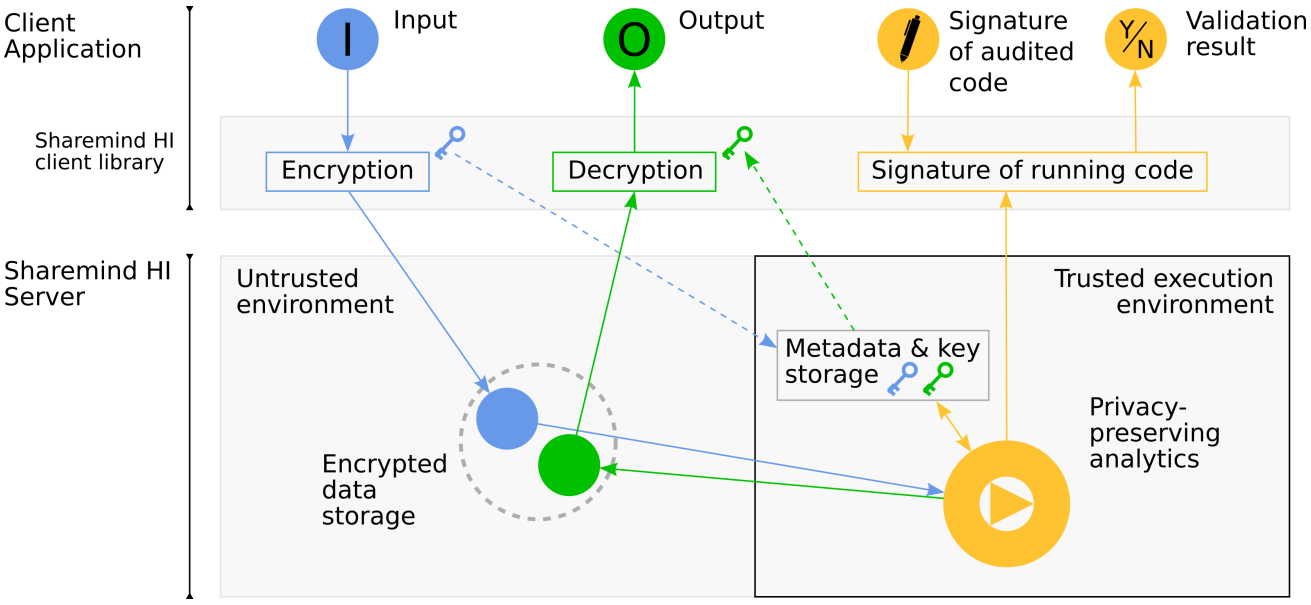

Figure 1. Sharemind HI security model

The typical life cycle of a Sharemind HI server solution is as follows:

1. Setup the Sharemind HI solution
2. Cryptographically approve the correctness of the server-side Sharemind HI solution
3. Upload input data
4. Run task enclaves
5. Download the result
6. Repeat steps 3 to 5 for continuous data analysis
7. Dispose of the Sharemind HI solution

Sharemind HI uses Intel® SGX to proceed with the following tasks: to verify the trustworthiness of the enclaves, to securely store encryption keys, to enforce authentication and access permissions, to protect the confidential data while it is being analysed, and more. Data can only move in or out of the Sharemind HI solution when the trustworthiness of the Sharemind HI solution has been cryptographically approved.

Information about activities on the server are stored in an audit log. The audit log can be downloaded and verified at any time.

3.2. Development of a Task Enclave

The analysis code of a Sharemind HI solution is compiled into task enclaves. Task enclaves for the Sharemind HI platform can be written with the Sharemind HI SDK which primarily targets the C++ programming language. The SDK offers a high level API which is transparently handling access control, data encryption as well as management of encryption keys. This allows the developer to concentrate on the algorithmic problem.

3.3. Development of a Client Application

Each Sharemind HI solution will be based on its unique data processing algorithms, and thus requires a custom data model with preprocessing and post-processing procedures. These procedures need to be implemented by a custom client-side application. To that extent, the

Sharemind HI platform provides three different client libraries for the following environments: Web (TypeScript), Native (C++) and scripting (CLI), none of which need a processor with support for Intel® SGX. The underlying technologies are sufficiently portable to reach even more environments.

3.4. Server Infrastructure Requirements

The Sharemind HI server requires a non-virtualized Linux server with an Intel® processor with Intel® SGX support. It does not matter whether the server runs on-premise or in the cloud. The Sharemind HI server can even run within an Ubuntu Docker container, as long as the Docker host is a non-virtualized Linux server.

To deploy security fixes, the Sharemind HI server and task enclaves need to be updated on average twice a year.

4. Security Model of Sharemind HI

4.1. Security goal

Confidential data is protected against any unauthorized access.

The security goal of Sharemind HI is as follows. When a stakeholder directs their confidential data into the Sharemind HI platform (where it will be encrypted), the data can only be accessed by those stakeholders and task enclaves which have the necessary permissions. No other party, such as the server administrator or malware, can access the unencrypted data.

4.2. Threat model

To describe the threat model, we first list the roles considered to be a threat to the security goal. We concentrate on attacks against confidentiality and integrity. For malicious attacks against availability, we assume the organisation has controls in place (e.g., contracts).

Coordinator

An individual interested in processing confidential data with Sharemind HI, and bringing the necessary stakeholders together to collaborate within a Sharemind HI solution.
A *maliciously acting Coordinator* has the following possibilities:

- **THR1** They may distribute malicious software and documentation disguised as Sharemind HI platform artifacts to the stakeholders (binaries, documentation, ...):
  - A malicious client library may silently forward plaintext data to a coordinator controlled location.
  - A malicious SDK may introduce arbitrary points of leakage into the task enclaves.
  - A malicious server may intentionally leak data encryption keys.
  - A malicious documentation may misguide stakeholders to follow insecure setup procedures or configure their client in an insecure way (distributing certificates and other identifiers over unauthenticated channels, ...).

Sharemind HI server administrator

An individual who has full access to the bare-metal machine and operating system which

runs the Sharemind HI server.
A *maliciously acting Sharemind HI server administrator* has the following possibilities:

- **THR2** They may intentionally misconfigure the Sharemind HI server.
- **THR3** They may use side channel attacks (running time, electromagnetic, power consumption, access patterns, cache timing, thread timing, eventual Intel® SGX vulnerabilities)
- **THR4** They may manipulate, add, drop or reorder messages between the Sharemind HI clients and Sharemind HI server, and between the enclaves of the Sharemind HI server itself.

Developers of the analysis code

An individual who implements or compiles the analysis code.
A *maliciously acting developer of the analysis code* has the following possibilities:

- **THR5** They may add intentional points of data leakage into the analysis code: augmenting the results, logging confidential data, introducing timing side-channels, etc. They maybe need to collude with other individuals to access the leaked data.

Enforcer

An individual who verifies that the Sharemind HI server is configured correctly and gives a cryptographically signed approval. Other stakeholders trust these signatures instead of validating the configuration themselves.
A *maliciously acting Enforcer* has the following possibilities:

- **THR6** They may intentionally approve a maliciously configured Sharemind HI server. Therefore, they need to collude with all other enforcers and the Sharemind HI server administrator.

Data Producer

An individual who is authorized to upload data to the Sharemind HI server.
A *maliciously acting Data Producer* has the following possibilities:

- **THR7** They may upload maliciously crafted data which exploits side-channel bugs in the analysis code of the task enclaves, possibly by colluding with other data producers.

A *misinformed Data Producer*:

- **THR8** They may publish their own confidential data.

Data Consumer

An individual who is authorized to download data from the Sharemind HI server.
A *misinformed Data Consumer*:

- **THR9** They may publish the analysis results to stakeholders that should not see these results.

Task Runner

An individual who is authorized to invoke the task enclaves.
A *maliciously acting Task Runner* has the following possibilities:

- **THR10** They may invoke the analysis code more often than agreed upon. They maybe need to collude with other individuals, e.g. with the administrator of the Sharemind HI server, to use this for data leakage.

Other individuals

An individual who has access to the communication channel between the parties. They are

less powerful than the Coordinator and the administrator of the Sharemind HI server.
Any other maliciously acting individual than the aforementioned individuals has the following possibilities:

- **THR11** They may manipulate, add, drop or reorder messages between the Sharemind HI clients and Sharemind HI server, and between the stakeholders.

Additionally: **THR12** A stakeholder may perform actions which were not agreed upon upfront with the Coordinator, and overall an individual may try to impersonate another stakeholder.

4.3. Requirements

REQ1
Stakeholders must use only Sharemind HI Client libraries and communicate only with an Sharemind HI Server (defeats THR1, THR8, THR9)

REQ2
Data Producers and Data Consumers must explicitly express their trust for the Enforcers (defeats THR6)

REQ3
Analysis code must not contain data leaks (defeats THR3, THR5, THR7, THR10)

REQ4
The server configuration must match the solution requirements (e.g. the role assignment must adhere to the principle of least privilege), and must be protected against manipulation (defeats THR2)

REQ5
An attacker shall not be able to extract confidential data from exchanged messages (defeats THR3, THR4, THR11)

REQ6
Ideally, any manipulation of communication channels should be detected and must not result in a data leak (defeats THR4, THR11)

REQ7
The Sharemind HI server must only process a request of a stakeholder if they have been authenticated and possess the necessary role (defeats THR12)

REQ8
An attacker shall not be able to extract confidential data from the persistent storage of the Sharemind HI server THR3)

REQ9
An attacker shall not be able to extract confidential data from working memory of the Sharemind HI server THR3)

4.4. Controls

Technical controls

- **TC1** The validity and measurements of an enclave can be cryptographically verified through Intel® SGX Remote Attestation and Local Attestation.
- **TC2** The measurements of an enclave, e.g. an enclave containing the analysis code,

can be derived locally from its source code.

- **TC3** The Sharemind HI client software only communicates with the enclaves whose measurements they trust.
- **TC4** The Sharemind HI server enclaves only share data with the enclaves whose measurements they trust.
- **TC5** Communication channels between clients and enclaves are protected through authenticated encrypted using ephemeral keys. Manipulating, adding, dropping or reordering messages will be detected.
- **TC6** The Sharemind HI server enclaves only process a request after the client has been authenticated and has the relevant permissions. Certificates for authentication and permissions for each stakeholder are declared in the Sharemind HI server configuration.
- **TC7** Data Producers and Data Consumers need to cryptographically express their trust for the Enforcers.
- **TC8** Data from Data Producers and from analysis enclaves is stored in encrypted form.
- **TC9** Long term encryption keys are stored in encrypted form using Intel® SGX Data Sealing.
- **TC10** The working memory of an enclave is encrypted by Intel® SGX.
- **TC11** The Sharemind HI server configuration becomes an immutable part of the protected persistent state after the initial invocation of the Sharemind HI server.

Organisational controls

- **OC1** Cybernetica distributes Sharemind HI platform components through secure channels.
- **OC2** Auditors audit the analysis code and inform the other stakeholders over a secure communication channel whether it is trustworthy according to the solution requirements or not.
- **OC3** Enforcers validate the remote Sharemind HI server configuration, including:
  - Stakeholder certificates: they receive the original certificates directly from the stakeholders over secure communication channels,
  - Enclave measurements which they compare against locally derived measurements from the audited (and trustworthy) analysis code.
- **OC4** Data Producers and Data Consumers check the Enforcers for trustworthiness, and if necessary take the role of an Enforcer themselves

Implementation of Requirements

- REQ1 is achieved with OC1
- REQ2 is achieved with TC7, OC4
- REQ3 is achieved with OC2, TC1, TC2, TC3, TC4, OC3
- REQ4 is achieved with OC3, TC11
- REQ5 is achieved with TC5
- REQ6 is achieved with TC5
- REQ7 is achieved with TC6, TC11
- REQ8 is achieved with TC8, TC9
- REQ9 is achieved with TC10

5. Further Features of Sharemind HI

5.1. Secure Configuration Upgrade

The configuration of the Sharemind HI Server, called Dataflow Configuration (DFC), contains crucial information about the stakeholder certificates and task enclave measurements. It needs to be approved by all enforcers when the Sharemind HI Server is started the first time. It is specially protected since a change to a task enclave measurement could leak sensitive data to a malicious task enclave. But task enclaves need to be updated from time to time for legitimate reasons, e.g. fixing bugs, security issues or adding new features. Other parts of the server configuration might need to be changed as well, like adding stakeholders, updating stakeholder certificates or changing roles. Hence Sharemind HI provides a mechanism to update the DFC in a controlled manner, where a new DFC is only becomes active when all enforcers have signed it.

5.2. Disaster Recovery

Sharemind HI uses Intel® SGX data sealing technology which is tightly coupled to the specific CPU on which it was used. This means, generating sealed data on one CPU and using it on another CPU will fail. To overcome this problem, Sharemind HI encrypts all sealing keys with a special asymmetric root key, whose private key component is secret shared across a configurable set of stakeholders. Secret sharing is used to keep the control over data recovery in the hand of many stakeholders during the migration period when there is no hardware protection from Intel® SGX. Stakeholders who want to participate in the disaster recovery create a special recovery key pair. The public recovery key is part of the server configuration and hence verified by the enforcers.

5.3. Dynamic End Users

The stakeholders of a given Sharemind HI solution are all listed within the server configuration. This is problematic for scenarios where the stakeholders want to provide the solution as a service to a larger audience, with users registering only at some arbitrary time in the future. This is solved in Sharemind HI with the concepts of *CA stakeholders* and *dynamic end users*. A CA stakeholder itself is listed in the server configuration, but they can sign additional certificates for the dynamic end users. The dynamic end users can then interact with Sharemind HI similar to how regular stakeholders interact with Sharemind HI.

6. Performance Specifications

6.1. Enclave Performance

In general, the code running inside of an Intel® SGX enclave can run nearly as fast as the same code would run outside of the enclave. However, not all workflows translate well to Intel® SGX and care should be taken when implementing the functionality inside the enclaves. For the current implementation of Intel® SGX, the overhead comes from two main sources: switching between the Intel® SGX modes and paging of encrypted memory.

Firstly, the switch between Intel® SGX modes occurs when the execution of the application moves from the untrusted part to the trusted part of the code or vice versa. While the cost of a single mode switch is small, the cost can add up if the applications has to do it very frequently. Secondly, paging of the encrypted memory is necessary because the Intel® SGX hardware has access to a limited amount of memory. If the application requests access to a trusted memory region that is not currently loaded, the current memory page has to be encrypted and moved out of the Intel® SGX memory region while the requested page will be moved in and decrypted. Because of this, running applications with a lot of random memory accesses on Intel® SGX technology can be slow.

In performance critical applications, both of these issues can be significantly reduced by careful application design.

6.2. Sharemind HI Performance

To verify the achievable speed under Sharemind HI, a sample application was created performing operations such as joining data columns, sorting and counting. We were able to process 2GB of data (100M records) in 10 minutes, and 20GB of data in 26 minutes. This result is not directly expandable to other Sharemind HI use cases because the performance of an application running in an Intel® SGX enclave is heavily dependent on the complexity of the application itself. However, it gives an idea of what kind of performance could be achieved for non-trivial applications.

7. References

- Intel® Software Guard Extensions - https://software.intel.com/en-us/sgx
- Intel® Attestation Service - https://api.trustedservices.intel.com/documents/sgx-attestation-api-spec.pdf

8. Document History

- Version 1.1: Adding information about further features of Sharemind HI.
- Version 1.0: Initial version.