Ryoan: A Distributed Sandbox for Untrusted Computation on Secret Data
Threat Model — 3 Perspectives

Users (Data Subjects)
- Doesn’t trust service providers
- Doesn’t trust platforms (e.g. OS)

Service Providers
- Trusts Intel SGX
- Trusts Ryoan
- Trust their own module not to leak its own secrets

Everyone
- Trusts Intel SGX
- Trusts Ryoan
I Have Trust Issues - TurboTax Example
I Have Trust Issues - TurboTax Example

User: WXYZ
Ryoan at 10000 ft

SGX Enclave

NaCl-based Ryoan Sandbox

User Data
Intel Software Guard eXtensions (SGX)

What’s SGX?

Intel Software Guard Extensions (SGX) offers hardware-based memory encryption that isolates specific application code and data in memory.

SGX allows user-level code to allocate private regions of memory, called enclaves, which are designed to be protected from processes running at higher privilege levels.

*Even when Operating System (OS) is compromised, application can still keep secrets.*
An enclave is a secure container that only contains the private data in a computation and the code that operates on it, which is isolated from the outside environment including privileged software including OS and hypervisor.
SGX Software (Remote) Attestation

SGX software (remote) attestation proves to the Ryoan user that she is using the service in a secure container hosted by trusted hardware before passing sensitive data.

Key exchange: A, $g^A$

Verification

$g^B$, Sign $AK(g^A, g^B, M)$

$M = \text{Hash(Initial State)}$

Ryoan Identification: All enclaves must have the same initial state

Shared key: $K = g^{AB}$

Trusted Platform

Attestation Key (AK)

Ryoan Container

Key exchange: B, $g^A$

*Private key embedded into the chip

*User use Intel's public key to check signature
Google’s Native Client (NaCl)

What’s Native Client?

“Native Client is a sandbox for running compiled C and C++ code in the browser efficiently and securely, independent of the user’s operating system”
Google’s Native Client (NaCl) (1)

Each SGX enclave contains a NaCl sandbox instance that loads and executes untrusted modules.
Google’s Native Client (NaCl) (2)

“Why is NaCl important?”

NaCl can impose restrictions on untrusted modules:

- Can only address module memory
- Limits (intercepts & replace) syscalls
- Cannot modify SGX state
Entities in Ryoan

- **Modules**
  - NaCl x86 binaries with application logic from service providers
    - Potentially malicious

- **Platforms**
  - Host Computation
    - Potentially malicious

- **Sandboxes**
  - Trusted code
    - Confine modules
    - Executed within enclaves
Trust no one but Ryoan & Intel SGX
Ryoan’s Goals

● Keep user data secret (protect data subject’s data -> confidentiality)
  ○ Without trusting software stack and or infrastructure in-placed

● Ability to process user’s confidential data in a distributed application through confined communication between different service providers
  ○ Prevent covert channels
  ○ Stop an untrusted application from intentionally and covertly using users’ data to modulate events like system call arguments or I/O traffic statistics
Chain of Trust

![Diagram showing a chain of trust between Intel Skylake, Ryoan, and Module.]
Chain of Trust (1)

Enclave
- Meta Data
- Hash
- Signature

Ryoan
- Hash
- Signature

Module

SGX
Hardware
Ryoan’s Distributed Sandbox

- Restricts accessible memory
  - Prevents module from copying secrets to other parts of memory
- Enforces encryption and any system calls are emulated in enclave’s memory
  - Prevents module from exploiting system calls to write out secrets

- Prevents module from reading secrets out of memory
Stateless Module Enforcement Between Reqs

- Initialization
- Read Data
- Process Input
- Write Output
- Destroy

- Prevent modules from colluding with users to steal data
How Does Ryoan Work? DAG (TurboTax Example)

- User data sent to TurboTax Process Input module
- TurboTax Process Input module sends user data to GCP Classifier Module
- GCP Classifier sends output to TurboTax Return Result Module
How Does Ryoan Work? TurboTax Back Again!

- TurboTax adds its tag to prevent the secrets from flowing back to user
- Secret data are encrypted
- Secret data are flushed after processing

User wxyz’s tag is added to secret data sent to TurboTax

Secure Channel
How Does Ryoan Work? TurboTax Back Again!

- GCP adds its tag to prevent the secrets from flowing back to TurboTax
- Secret data are encrypted
- Secret data are flushed after processing

![Diagram showing the process of Ryoan working with TurboTax and Google Cloud Platform.]
How Does Ryoan Work? TurboTax Back Again!
How Does Ryoan Work? TurboTax Back Again!
How Does Ryoan Work? TurboTax Back Again!
How Does Ryoan Work? TurboTax Back Again!
How Does Ryoan Work? TurboTax Back Again!

Enclave
Ryoan Sandbox
TurboTax
xyz
Enclave
Ryoan Sandbox
TurboTax
xyz
Enclave
Ryoan Sandbox
Google Cloud Platform
Enclave
Ryoan Sandbox
TurboTax
xyz
Tax Credit Result
Secure Channel
Question 1

“If Intel’s SGX is not used in Ryoan, what guarantees does Ryoan provide?”
Answer 1

1. Without SGX, privileged software (OS, hypervisor, etc.) will have access to module memory
   a. Secret data could be leaked / unauthorized access

2. No code / data verification between the service user and the remote container
   a. Identities of containers will no longer exist
   b. User have no idea whether the module have been tampered or modified
“If Google’s NaCl is not used in Ryoan, what guarantees does Ryoan provide?”
Answer 2

Without NaCl, Ryoan will lose three security properties enforced by NaCl!

1. Untrusted modules can address not only module memory but also memory that doesn’t belong to themselves

2. Ryoan will not be able to intercept syscalls from these modules which may be malicious

3. The restriction that this module cannot modify SGX state will also be lifted
Limitations

● Slow performance
  ○ Each module running within Ryoan can only process 1 user data at any point in time

● Fixed execution - Services that are called are defined ahead of time -> DAG

● Applications might require custom libraries, however these libraries does not exist in Ryoan’s libc

● Memory limitations
  ○ Module(s) which requires large memory usage cannot be loaded in as a single module

● Intel processors hardware limitations - compromises Ryoan’s security goals
  ○ SGX page faults, cache timing, address bus monitoring, processor monitoring
Thank you for the kind attention!
Reference


Intel introduces three Skylake "R" class processors - NotebookCheck.net News

Native Client

Conference Slides