A Case for Protecting Computer Games With SGX

Erick Bauman and Zhiqiang Lin

System and Software Security (S^3) Lab
The University of Texas at Dallas

December 12th, 2016

Outline

Outline

Computer Games

- Large industry, market value of tens of billions
- Popular games have millions of players

Cheat Prevention

- **Cheating in multiplayer games** serious concern for developers
- Small percentage of players can ruin experience for majority

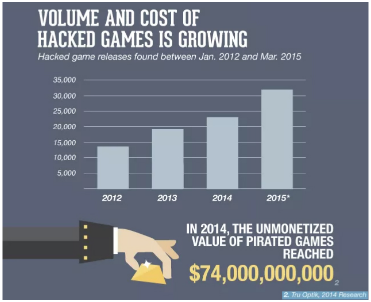
Cheat Prevention

- A million-dollar industry
- Difficult to defend against
  - Cannot trust client machines
  - Server-side integrity checks often have high overhead

BE A COUNTER-STRIKE GOD
FOR ONLY $10.95 PER MONTH!
See through walls!
Use an aimbot to score killer headshots!
You won't BELIEVE these hacks!

DRM

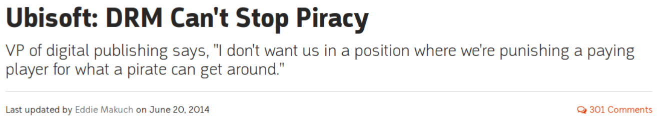
- Easy data duplication makes sharing applications trivial
- Many companies have strong interests in copy protection
- Piracy often costs billions in lost sales



DRM: preventing circumvention of protection is hard

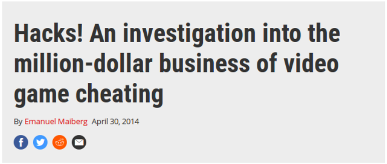
- Usually requires a trusted component on user's machine
- Trusted component is protected by complex obfuscation, often quickly reverse-engineered
- Secrets are often too easily extracted without a way to truly secure them

Background

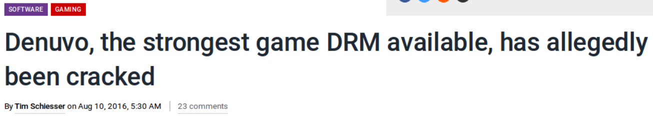

Ubisoft: DRM Can't Stop Piracy

VP of digital publishing says, "I don't want us in a position where we're punishing a paying player for what a pirate can get around."

Last updated by Eddie Makuch on June 20, 2014 — 301 Comments

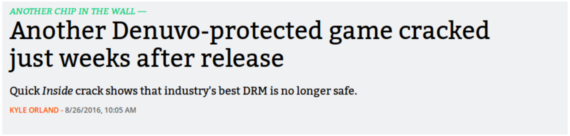

Hacks! An investigation into the million-dollar business of video game cheating

By Emanuel Maiberg April 30, 2014


SOFTWARE GAMING

Denuvo, the strongest game DRM available, has allegedly been cracked

By Tim Schiesser on Aug 10, 2016, 5:30 AM | 23 comments


ANOTHER CHIP IN THE WALL —

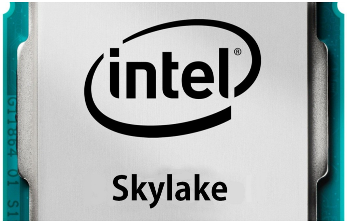
Another Denuvo-protected game cracked just weeks after release

Quick *Inside* crack shows that industry's best DRM is no longer safe.

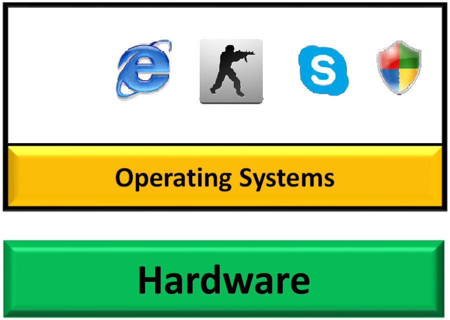
KYLE ORLAND - 8/26/2016, 10:05 AM

Intel SGX

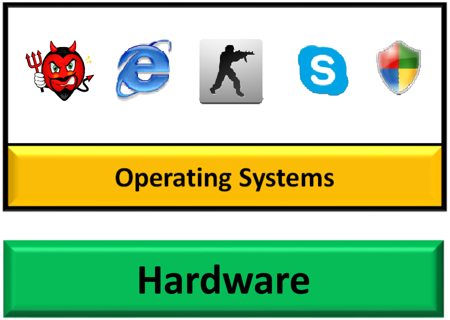
- SGX's secure enclaves provide strong guarantees to protect applications
  - Isolated execution environment
  - Contents unreadable by machine owner
  - Protection enforced by hardware



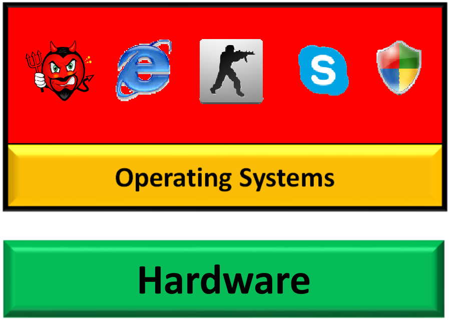
Why Intel SGX



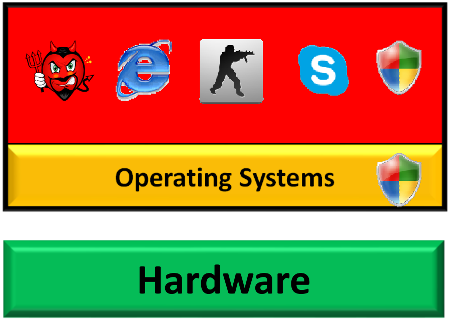
Why Intel SGX



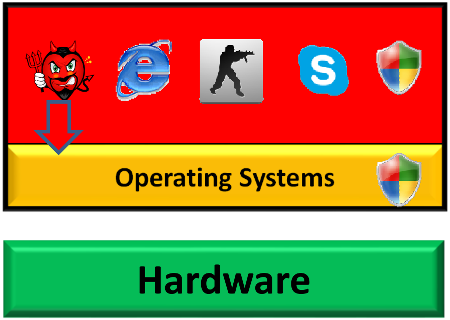
Why Intel SGX



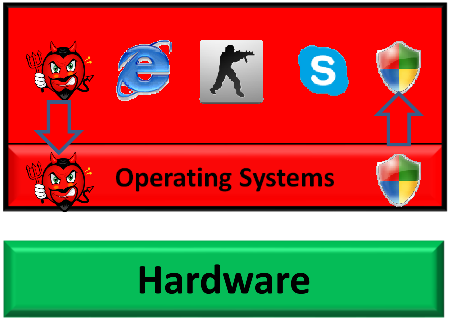
Why Intel SGX



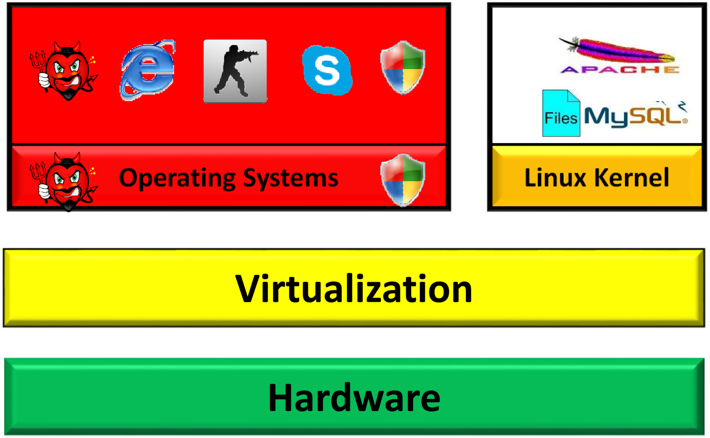
Why Intel SGX



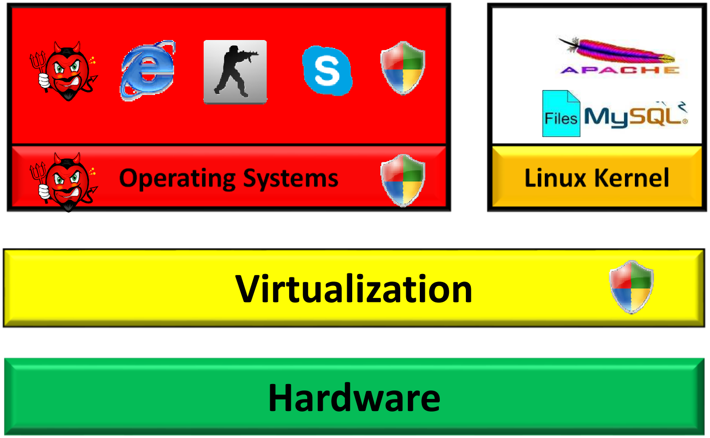
Why Intel SGX



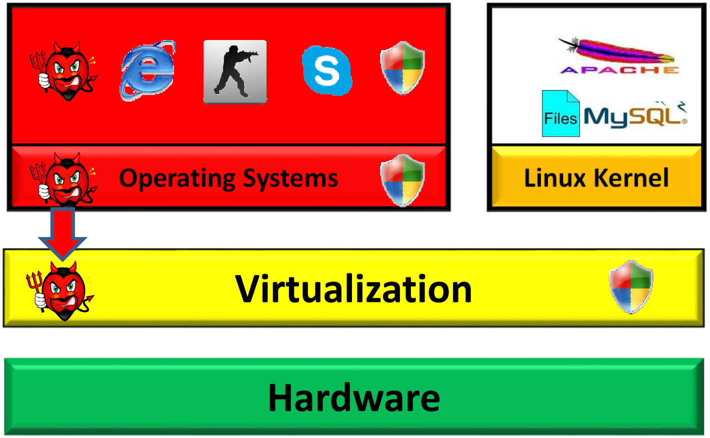
Why Intel SGX



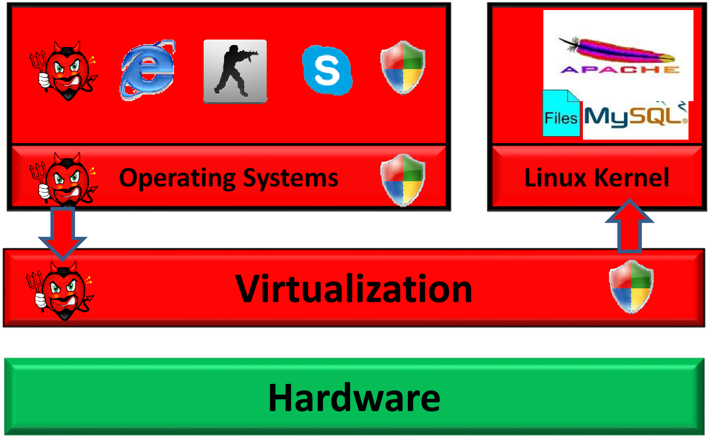
Why Intel SGX



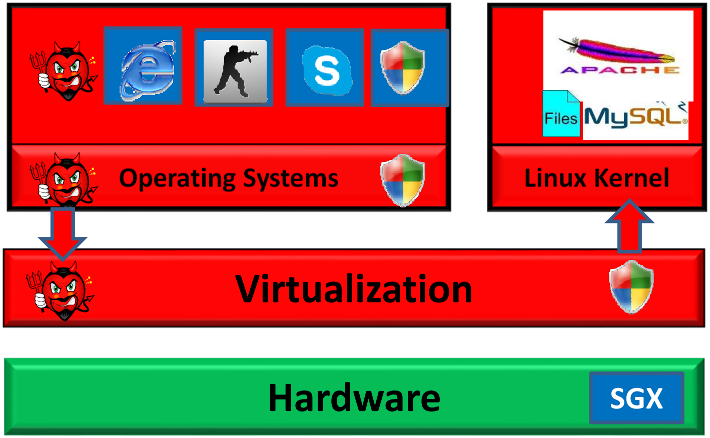
Why Intel SGX



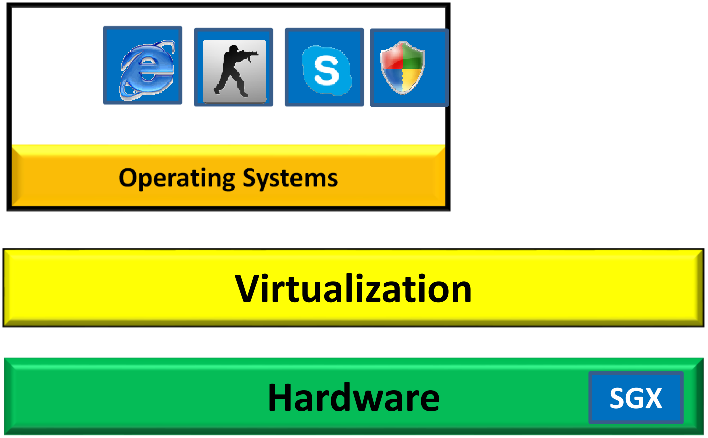
Why Intel SGX



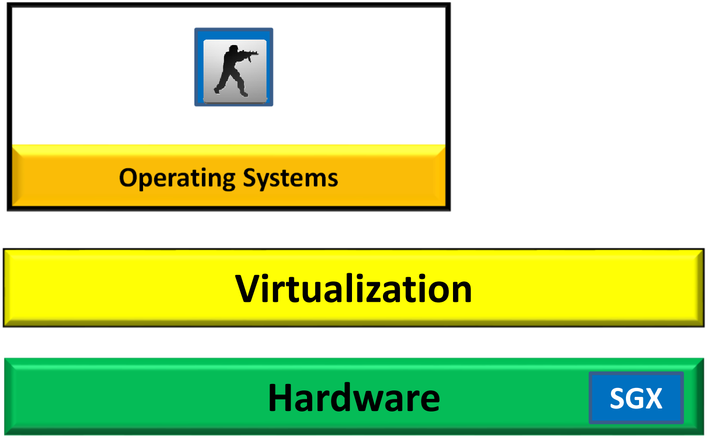
Why Intel SGX



Why Intel SGX



Why Intel SGX



Key SGX Features of Interest

Outline

Scope and Assumptions

Scope: Computer Games

- Multiplayer games for **cheat prevention**
- Single and multiplayer games for **DRM**

Scope and Assumptions

Scope: Computer Games

- Multiplayer games for **cheat prevention**
- Single and multiplayer games for **DRM**

Assumptions and Threat Model

- An attacker may have **full control over all software except for trusted enclaves**
- Attacker may access all memory, but not the processor
- We assume SGX itself is secure

Protection Model

Integrity: Crucial for Cheat Prevention

Data Integrity

- Prevent disallowed modifications to data
- Protect code that does modify data
- Provide limited interface for modifying data

Code Integrity

- Prevent modifications to crucial code, e.g. validation code
- Move necessary code to enclave

Protection Model

Confidentiality: Crucial for DRM

Data Confidentiality

- Any data decrypted inside enclave remains hidden
- If data must be shown to user, it may potentially be extracted from memory without secure I/O
- If code that touches data can reside entirely inside enclave, data can remain hidden

Code Confidentiality

- More challenging than code integrity
- Enclave code can be read before enclave is instantiated
- Code must be dynamically decrypted in enclave at runtime
- Can result in complete black box for user

Protection Model

Examples

	Integrity	**Confidentiality**
Data	Game State: Score, lives, orientation, map inventory items player position	Media Content: sounds, textures 3D models configuration data
Code	Integrity Checks: Velocity Checks Collision Detection	Game Logic: Algorithms Scripts

Desired Properties for Protected Content

Isolated

- Enclaves prohibit certain instructions, e.g. system calls
- Enclave code must be isolated from the application code
- Data sent across enclave boundary must be copied
- Presents a challenge to port existing applications to SGX!

Desired Properties for Protected Content

Isolated

- Enclaves prohibit certain instructions, e.g. system calls
- Enclave code must be isolated from the application code
- Data sent across enclave boundary must be copied
- Presents a challenge to port existing applications to SGX!

Crucial

- Enclaves have a limited amount of memory available
- An enclave too large for EPC will hurt performance
- The larger the code in enclave, the greater the risk of vulnerability or side channel

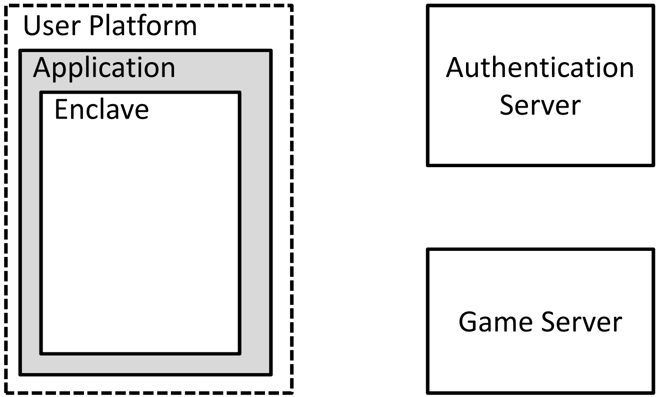
Outline

Protecting Integrity

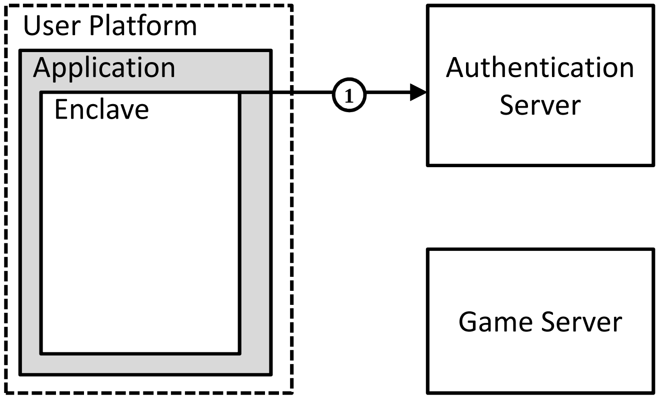
Key Ideas

- Multiplayer games must have one or more game servers
- Server-side integrity checks may be expensive
- SGX allows a single, one-time check of enclave integrity
- After attestation, all signed or encrypted messages from the enclave can be trusted without further checks
- Code and data inside enclave can therefore be trusted

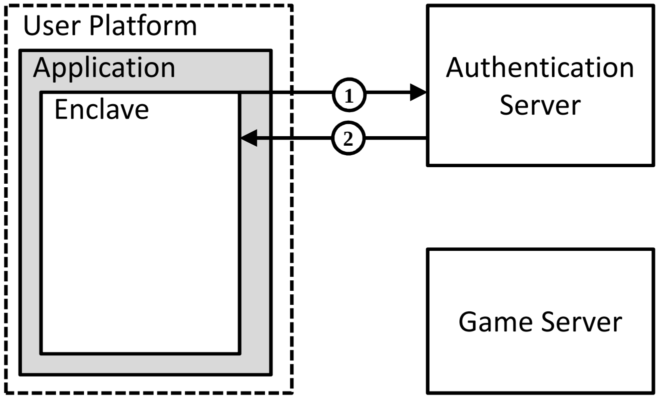
Protecting Integrity



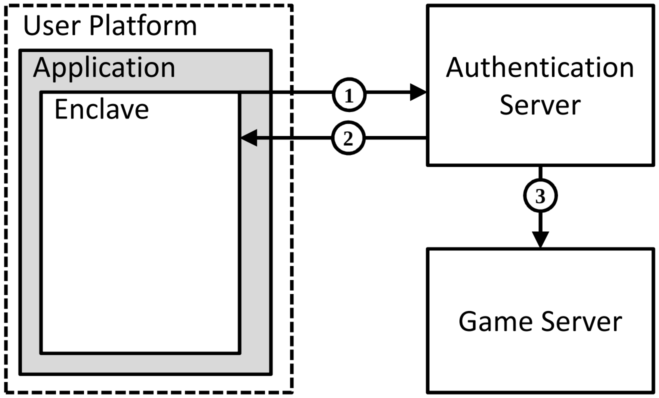
Protecting Integrity



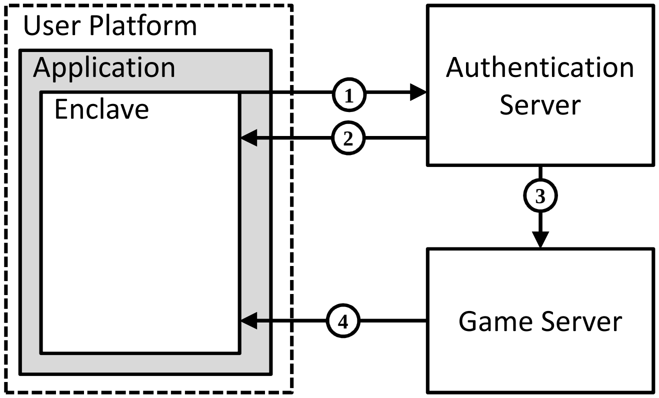
Protecting Integrity



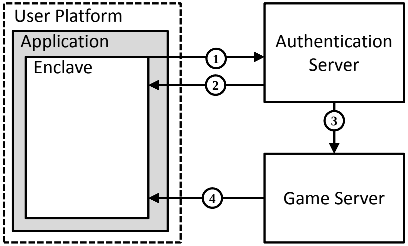
Protecting Integrity



Protecting Integrity



Protecting Integrity: Recap



Detailed Steps

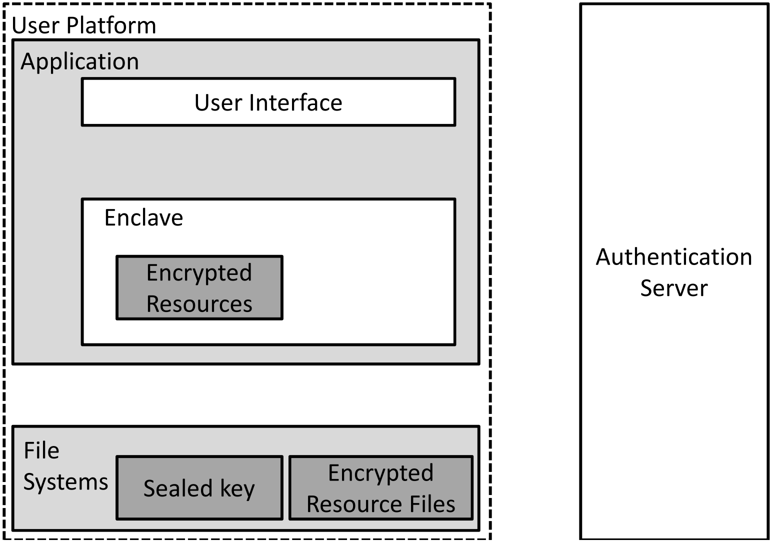
1. Start Remote Attestation
2. Verify Enclave
3. Share Credentials
4. Enclave Communicates with Game Server

Protecting Confidentiality

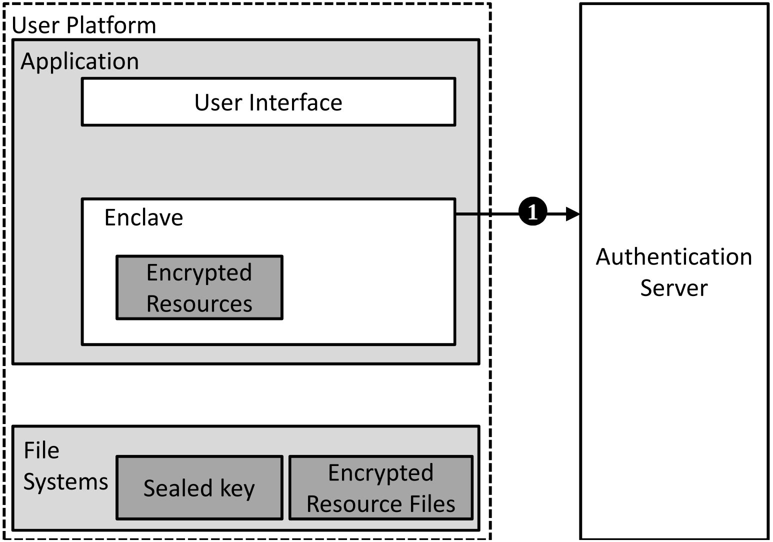
Key Ideas

- Content can be protected by encryption
- All data decrypted inside enclave is secure
- Key to decrypt content can be withheld until proof of purchase is given
- Authentication server gives decryption key only after successful attestation and license key is given
- After initial license check, enclave can seal key to allow resource decryption without contacting server

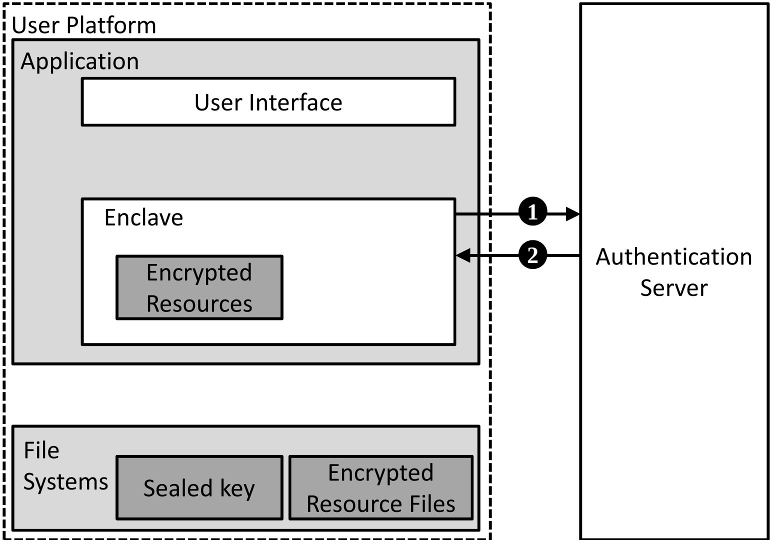
Protecting Confidentiality



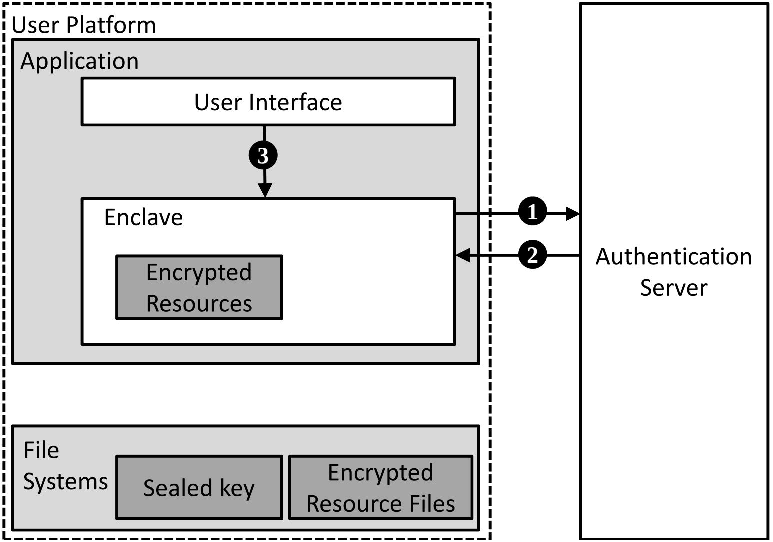
Protecting Confidentiality



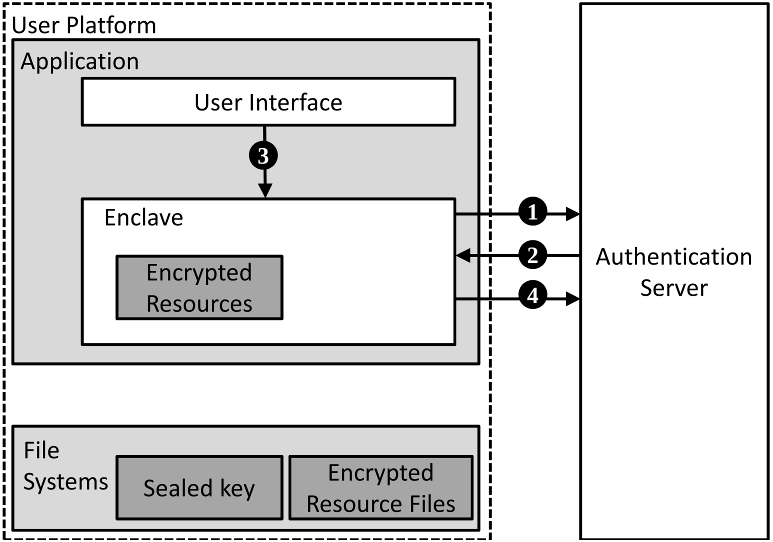
Protecting Confidentiality



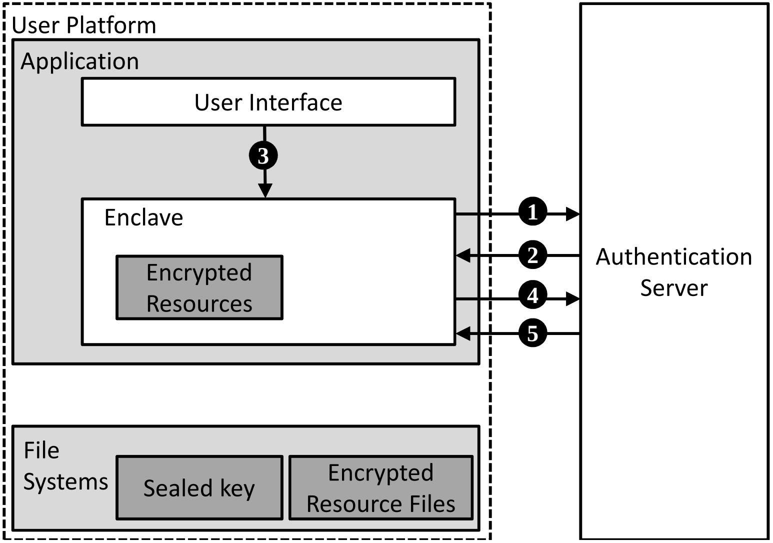
Protecting Confidentiality



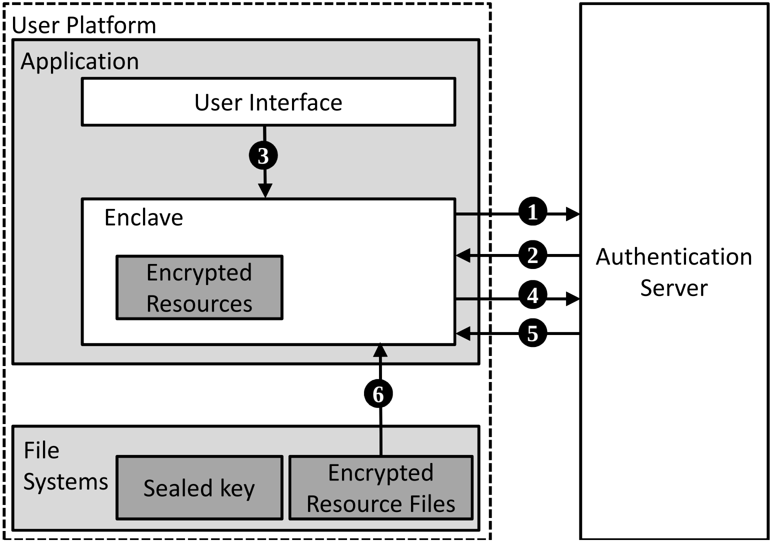
Protecting Confidentiality



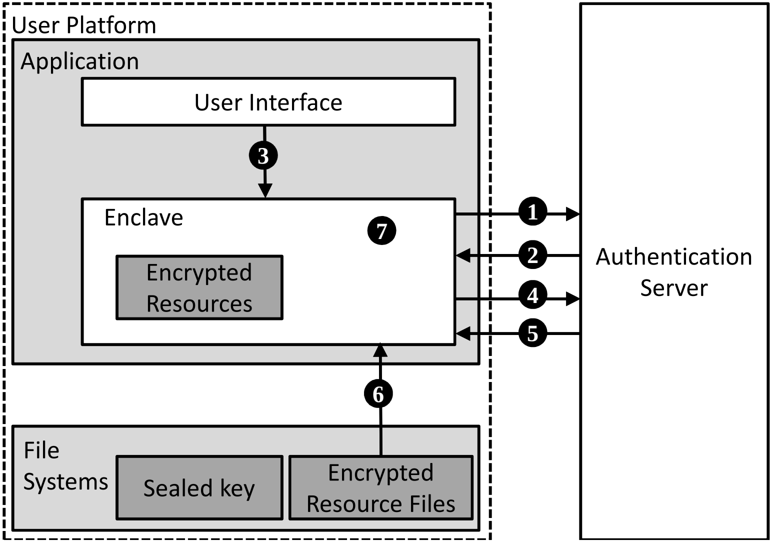
Protecting Confidentiality



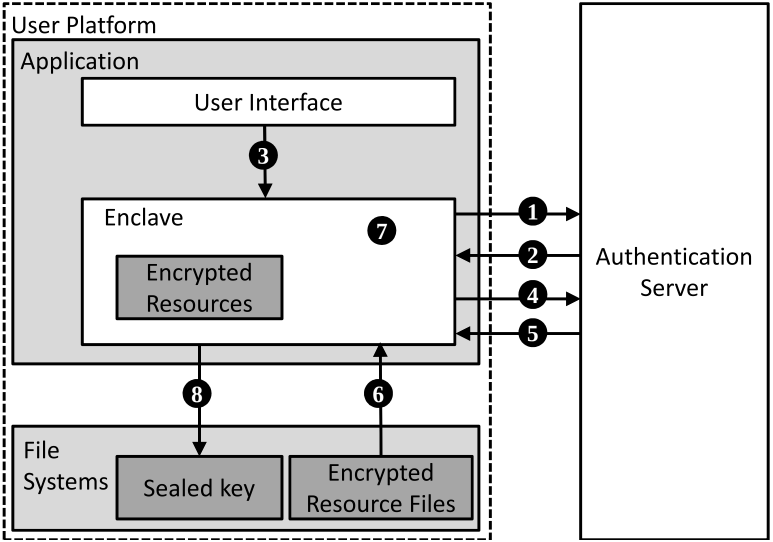
Protecting Confidentiality



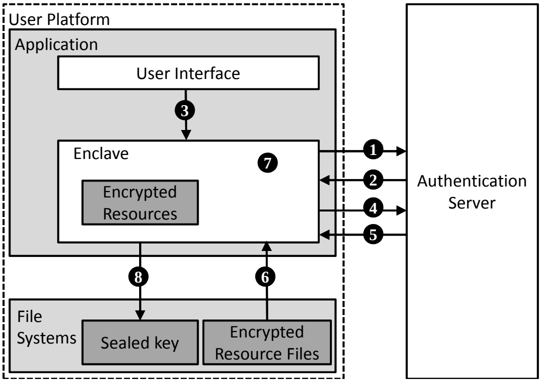
Protecting Confidentiality



Protecting Confidentiality



Protecting Confidentiality: Recap



Detailed Steps

1. Start Remote Attestation
2. Verify Enclave
3. Retrieve License Key
4. Send License Key
5. Receive Decryption Key
6. Retrieve Encrypted Assets
7. Decrypt Assets
8. Seal Decryption Key

Outline

Challenges

- Each game requires protection of different content (i.e., **Protection is game specific**)
- **Partitioning is difficult**
  - Existing games not designed with isolated component
  - Many code dependencies
  - Can lead to too much code in enclave
  - Difficult to balance enclave size with securing enough code and data
- Many assets will be leaked due to **lack of secure I/O**

Objectives

Port Real Game to SGX

Open-source game Biniax2, consisting of over 3500 lines of C

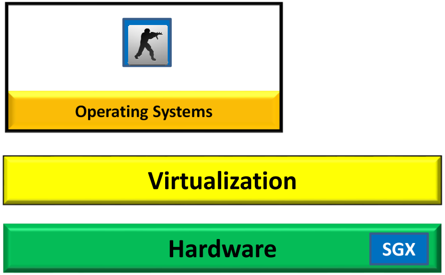
Objectives

Applying Our Framework

Focus on **DRM protection mechanisms** since game does not support networked multiplayer

Protecting Assets

Prevent assets from being loaded until encryption key is provided



Modifications

- Partitioned application into trusted and untrusted components
- Modified asset handling code to load encrypted assets
  - 923KB of images
  - 160KB of sound effects
  - 14KB of text
- Provided proof-of-concept confidentiality protection for assets

Performance

Metric	Biniax2	SGX-Biniax2	Increase
Lines of Code	3540	4326	22.20%
Initialization Time (ms)	141.58 ± 4.23	243.59 ± 4.11	72.05%
Binary Size (bytes)	35038	38353	9.46%
Asset Size (bytes)	1084486	1097259	1.18%

Table: Comparison of several metrics between the original Biniax2 game and our modified version that we ported to SGX.

Performance

Metric	Value
Lines of Code in Enclave	580
Enclave Size (bytes)	100425
Enclave Initialization (ms)	53.22 ± 4.21
Assets Encrypted	29

Table: Statistics for our modified SGX-Biniax2.

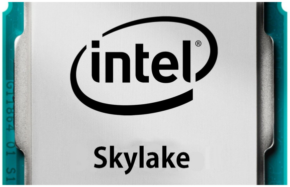
Future Work

- Encrypt secrets that never need to leave enclave
- Fully demonstrate attestation, sealing, and unsealing
- Perform case study for cheat prevention
- Further analyze security implications of enclave applications and how to prevent implementation vulnerabilities

Outline

Conclusion

- SGX provides an excellent opportunity for protecting games and applications
- We demonstrated a general framework that takes a first step in **using SGX for DRM and cheat prevention**
- We performed a case study showing the feasibility of our approach



Thank You

Q&A