DIGITAL RIGHTS MANAGEMENT IN A 3G MOBILE PHONE AND BEYOND (2003)

PRESENTATION : 김용현(2017-22945)

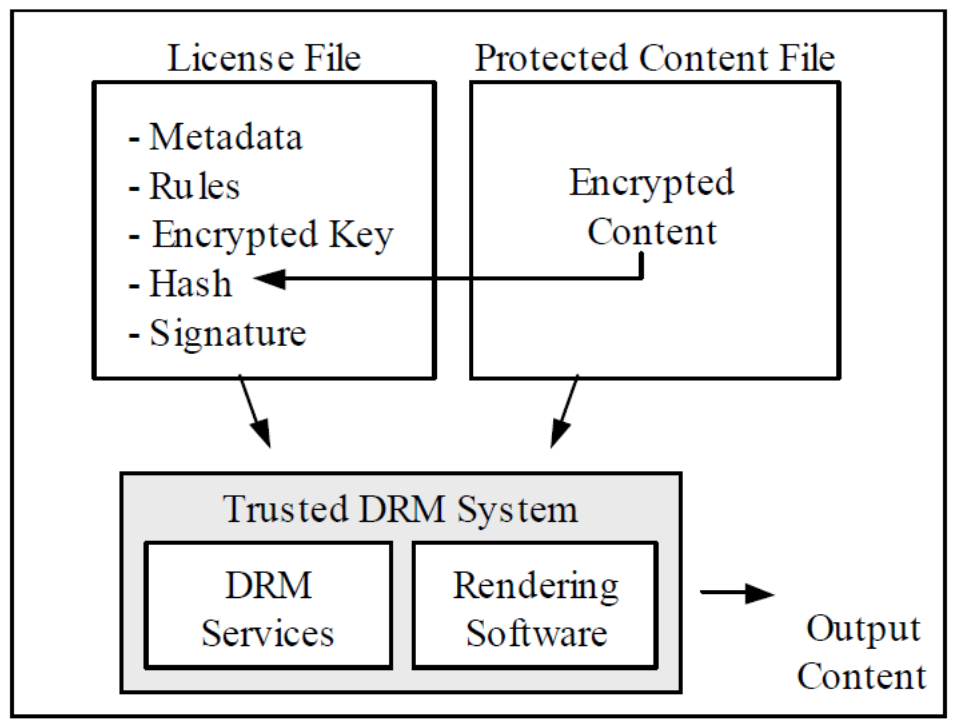
INTRODUCTION

- Data Transition with mobile device is accelerating
- So, market is growing in this field
- Copying data too chip -> piracy is increasing
  - $12 Billion loss due to piracy in 1999
  - DRM is needed

DRM(DIGITAL RIGHT MANAGEMENT)

- There are many long description in the paper. But in Summary…
- Without rights, you can not access(render, transport, etc..) contents
  - If it charges fee, you must pay.

DRM-FUNDAMENTAL APPROACH



DRM-RENDERING ORDER

1. The trusted rendering software sends the encrypted contents and the corresponding license to DRM.
2. Verify license
3. Verify hash of the contents
4. Decrypt protected contents
5. Send decrypted contents to the rendering software
6. Rendering software “runs” decrypted contents

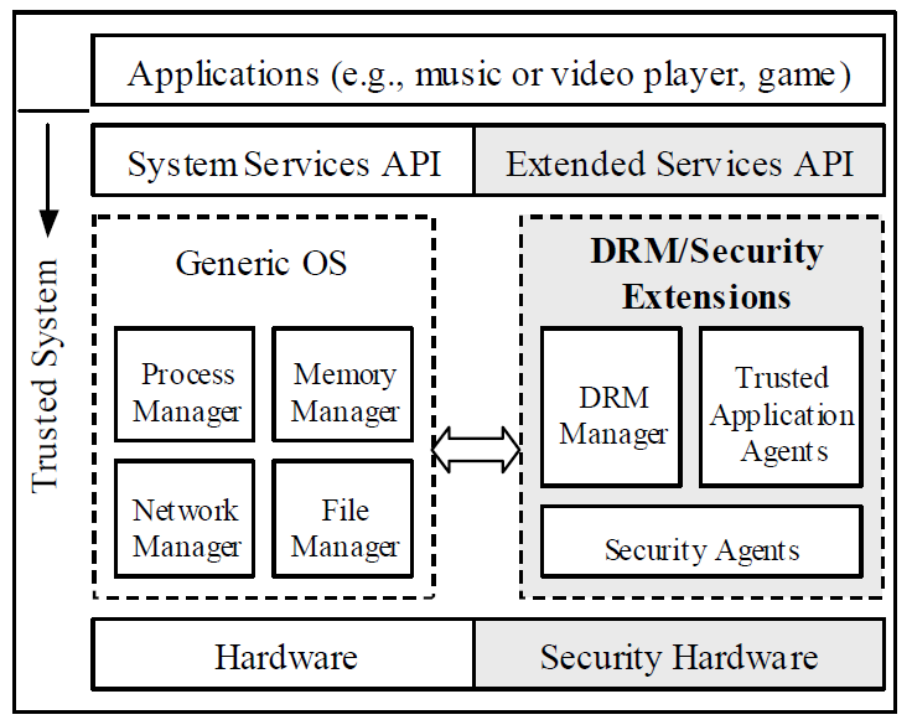
DRM-PREVIOUS APPROACHES(SCHNECK)

1. replace the I/O elements of the OS with modules that contain access control mechanisms.
2. “hyperadvisor” - between the OS and the hardware.
   - invoke the DRM system and special software and hardware would complete the operation.

DRM-APPROACH IN PAPER

- Extended OS - to support DRM functionality
  - Divide into two
  - “user mode” - GUI and High level logics etc…
  - “privileged mode” - access to system data and resources.

DRM-APPROACH IN PAPER



DRM MANAGER

- Works with security agents
- authenticate licenses and content
- parse and enforce usage rules
- access a secure DRM database
- provide decrypted content to a trusted application agent

DRM MANAGER - AUTHENTICATE LICENSES AND CONTENT

- This check will typically require that the cryptographic hash of the license file be computed and that a digital signature be verified.
- if licenses and content are packaged to include a hash table.
  - Distributed hash verification
  - Good for embedded processor like mobile phone
- Check license and content files originate from valid sources and whether they have been modified.

DRM MANAGER – ENFORCE RIGHT

- Application ask to DRM Manager
  - About perform an action(play, display, copy, etc..) on the content.
- Fundamental types of rights
  - Render, Transport, Derivative
- use a secure database for track special events like
  - Sometimes the license will stipulate an additional event for performing an action
  - Ex) payment needs to be made, a play count needs to be decremented.
- to enforce the usage rules
  - Access device's credentials

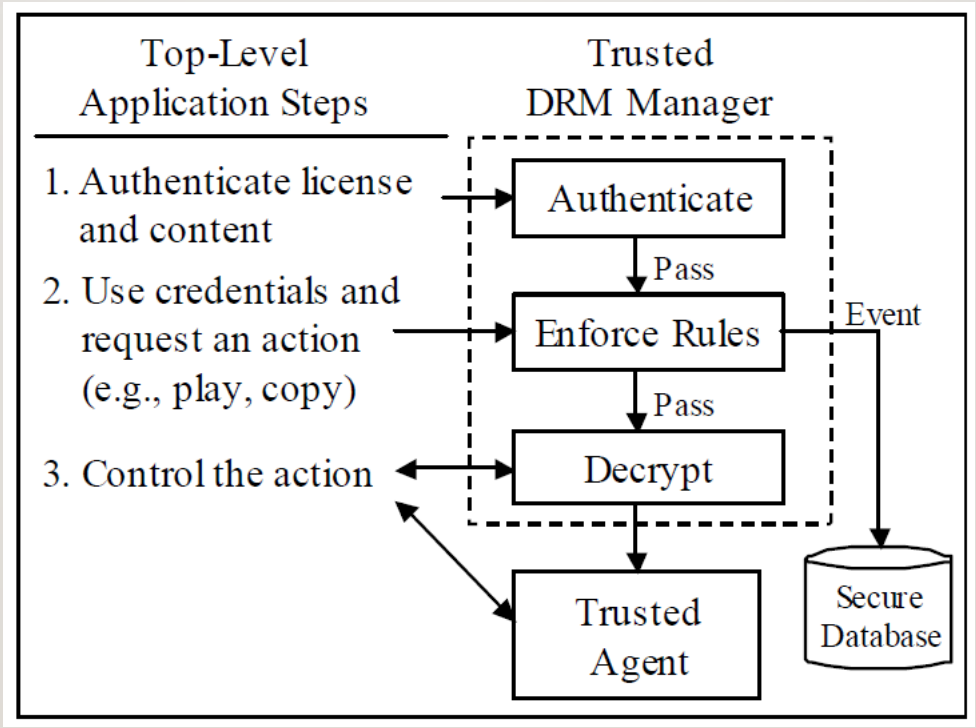
DRM MANAGER – DECRYPT CONTENT

- High level applications may not be able to access directly decrypted contents
  - not a trusted OS and may not have permissions
- DRM Manager sends decrypted contents to a trusted application (such as rendering software).

DRM MANAGER – DECRYPT CONTENT

- High level applications may not be able to access directly decrypted contents
  - not a trusted OS and may not have permissions
- DRM Manager sends decrypted contents to a trusted application (such as rendering software).

DRM MANAGER – EXAMPLE



TRUSTED APPLICATION AGENTS

- Part of the extended OS.
- Support the ability of applications to access and manipulate decrypted content.

TRUSTED APPLICATION AGENTS-RENDERING AGENTS

- Provide applications the render DRM-protected content
  - Ex) music player, a picture viewer
- Tightly coupled to the top-level application.
- Application loader
  - responsible for enforcing usage rules prior to executing a previously installed application

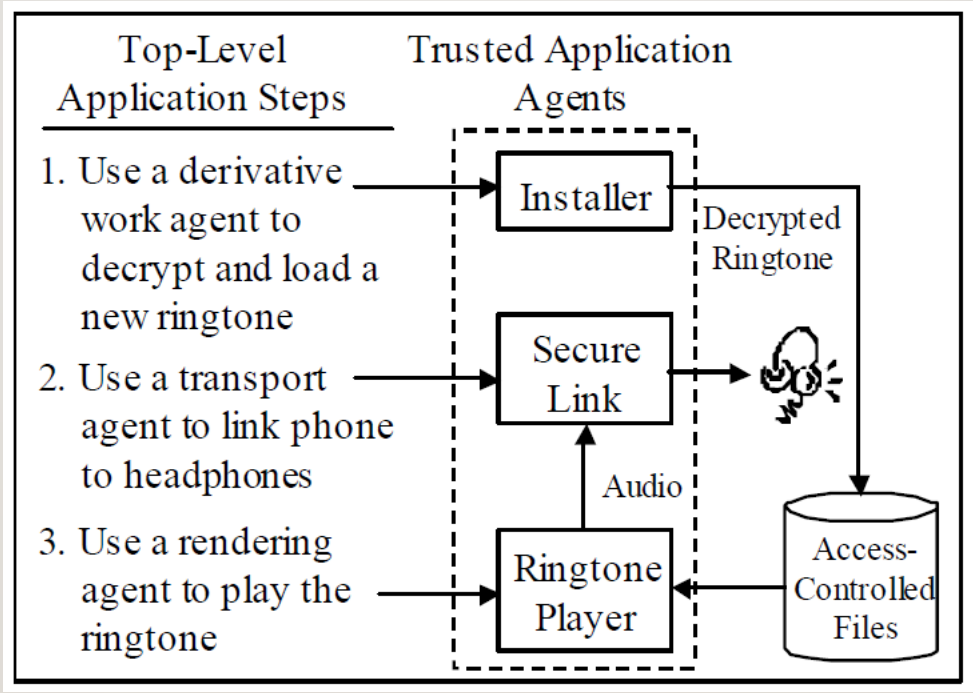
TRUSTED APPLICATION AGENTS-TRANSPORT AGENTS

- Provide services that move content from one location to another
  - Ex) email attachments, messaging services etc
- Because it has to carry decrytped contents,
  - transport agents also need to be trusted

TRUSTED APPLICATION AGENTS-DERIVATIVE WORK AGENTS

- extract and transform protected content into a different form.
  - copy of a digital item might have different rights than the original.

TRUSTED APPLICATION AGENTS-EXAMPLE



SECURITY AGENTS

- Security-related functions that are commonly needed in all DRM systems.
  - secure memory file management
  - cryptographic operations
  - key management

SECURITY AGENTS - MEMORY AND FILE MANAGEMENT

- A DRM system needs to ensure that access to memory and files can be controlled.
  - You need to be able to access only what you want.
- 3 security functions related to memory and file management.
  - access-controlled file system
  - Secure memory system
  - memory separation system

ACCESS-CONTROLLED FILE SYSTEM

- Use case
  - provides is the storage of digital content that is no longer encrypted.
  - store a secure database
- Requirement
  - Files are assigned ownership attributes that specify which trusted agents can access the files.
  - Tampering of the ownership attributes can be detected
  - Files are optionally encrypted
  - Files that are not encrypted must be physically located within the phone.

MEMORY SEPARATION SYSTEM

- We want to ensure that when a trusted operation is running, untrusted operations cannot eavesdrop on the memory being used.

SECURE MEMORY

- DRM system there is critical data that should never be allowed to leak out of the system.

SECURITY AGENTS - CRYPTOGRAPHIC OPERATIONS

Operation	Time
Hash of a license (5KByte)	SHA1: 3 ms
Verify license signature	RSA(1): 100 ms ECC(2): 150 ms
Decrypt content key	RSA(1): 1,800 ms ECC(2): 90 ms
Decrypt content (2 Kbyte)	AES(3): 1.6 ms

(1) 1024-bit RSA with CRT (2) WTLS Curve 3 (3) 128-bit key

Figure 5. Typical execution times for processing DRM-protected content using software implementations of RSA, ECC, SHA-1, and AES on a 16 MHz ARM7 microprocessor. The above data shows that if hardware is not available, ECC is much better suited for wrapping content keys.

- provide access to symmetric and public-key cryptographic functions.
- AES : Protected content is encrypted using a symmetric-key algorithm
- SHA-1 : binding between content and licenses is done with a hash algorithm
- RSA, ECC : public key operation

SECURITY AGENTS - KEY/CERTIFICATE MANAGER

- software module responsible for securely handling a database of the phone's credentials
  - private keys, public keys, certificates, and identification numbers.

DRM CREDENTIALS

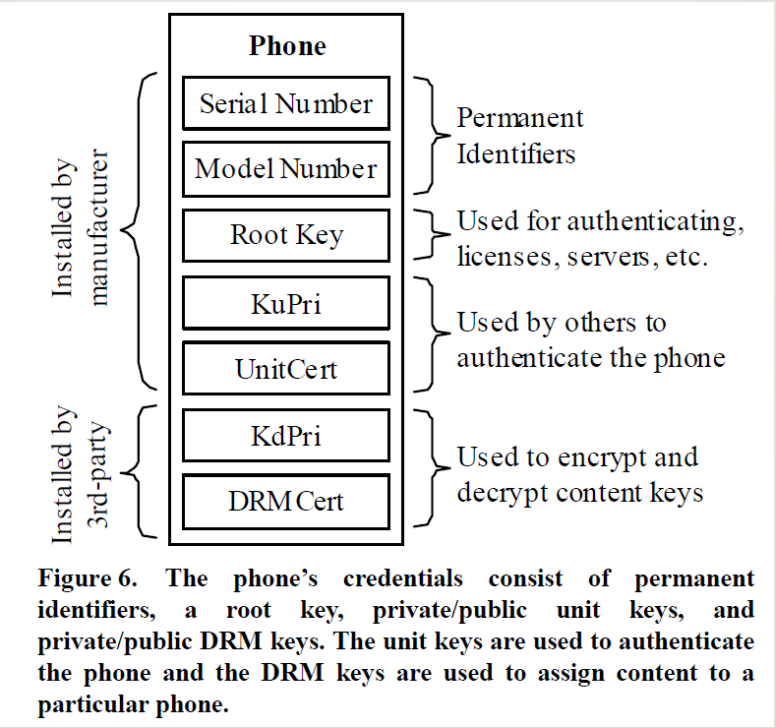


Figure 6. The phone's credentials consist of permanent identifiers, a root key, private/public unit keys, and private/public DRM keys. The unit keys are used to authenticate the phone and the DRM keys are used to assign content to a particular phone.

DRM CREDENTIALS - SERIAL AND MODEL NUMBERS

- SN : an unchangeable number that unambiguously identifies the phone.
- MN : a number that unambiguously identifies the hardware and software version of a phone.

DRM CREDENTIALS -PRIVATE KEYS AND CERTIFICATES

- *KuPri :* the phone's unique private key
- *UnitCert :* a certificate that certifies the corresponding public key (*KuPub*).
- *KuPri* and *UnitCert* should be used for establishing secure-authenticated channels to a phone
- *KdPri :* also a unique private key
- *DRMCert :* also a certificate that certifies the corresponding public key (*KdPub*).

DRM CREDENTIALS – SHORT-LIVED CERTIFICATES

- valid for only a limited time
- A device's certificates can have expiration dates
- Time and Date source must be reliable.

FAMILY DOMAIN

- Users want to write their own contents on all their devices.
- Register all your devices in the domain(Domain Autority)
- Devices inside the domain have full access to the content and devices outside the domain do not.

FAMILY DOMAIN

- In our "Family Domain" system, portable devices are assigned to a particular domain by registering with the DA.
- JOINED : device registers into a domain
- LEAVE : canceling its registration
- You can also give someone in your other family a password to get into the Domain.
- Benefits
  - Track abusive activities by tracking overly active activities
  - Track joining and leaving to prevent abnormal behavior
  - You can also give someone in your other family a password to get into the Domain.