적응적 보안등급을 이용한 컨텐츠 암호화 모듈 설계 및 구현

김환조, 서정철, 정목동
부경대학교 컴퓨터공학과

Design and Implementation of a Content Encryption Module Using Adaptive Security Level

Hwan-Jo Kim, Jung-Chul Seo, Mokdong Chung
Dept of Computer Engineering, Pukyong National University

요 약

컴퓨팅 환경이 유비쿼터스 환경으로 변해가면서 다양한 컨텐츠와 다양한 디바이스들이 등장하게 되었고, 디지털 컨텐츠를 보호하기 위해 DRM(Digital Rights Management) 기술이 적용된 서비스가 제공되고 있다. 그러나 현재 DRM 기술의 디지털 컨텐츠 보안 정책은 일정한 키 길이에 동일한 암호 알고리즘을 사용함으로써 비효율적이고, 사용자의 다양한 요구를 만족시키지 못하고 있다. 본 논문에서는 디지털 컨텐츠에 보안 정책을 효율적으로 적용하고 디바이스의 성능과 디지털 컨텐츠의 가치에 따라 의사 결정 방법인 MAUT(Multi-Attribute Utility Theory) 알고리즘을 이용하여 최적의 보안 등급을 동적으로 결정하는 컨텐츠 암호화 모듈을 설계하고 구현한다.

1. 서론

인터넷과 통신의 발전으로 인해 디지털 컨텐츠의 유통 및 이용이 증가되었지만 디지털 컨텐츠의 특성인 불법 복제와 손쉬운 유통으로 저작권 침해 문제가 발생하고 있다. 이를 해결하기 위해 DRM (Digital Rights Management)[1,2]기술이 사용되고 있다.

DRM은 디지털 컨텐츠가 생성될 때부터 배포, 이용될 때까지의 전체 생명주기에 걸쳐 적용되며, 각각의 사용자가 사전에 정해진 조건을 만족해야만 이용할 수 있는 장치로서, 컨텐츠의 자유로운 유통은 허용하지만 불법사용은 철저히 막는 기술로 정의 할 수 있다. DRM은 MS[3], Intertrust[4]등과 같은 여러 개발업체에서 활발히 연구되고 있다. 현재의 DRM에서는 사용자들에게 보호된 컨텐츠를 제공하기 위해서 디바이스 성능이나 컨텐츠의 가치를 고려하지 않고, 일정한 키 길이에 동일한 암호 알고리즘을 보안 정책으로 사용하고 있다. 그러나 디지털 컨텐츠는 가격, 사용기간, 품질에 따라 가치 차이가 있고, 디바이스의 성능의 차이에 따라 컨텐츠를 암·복호화하는 시간과 시스템 리소스 점유율이 다르다. 그러므로 디바이스의 성능 및 컨텐츠의 가치에 따라 적절한 보안 정책을 동적으로 결정하는 알고리즘이 필요하다.

본 논문에서는 디지털 컨텐츠를 이용하는 디바이스의 성능과 컨텐츠의 가치에 따라 보안 정책을 동적으로 결정할 수 있는 알고리즘을 개발하고, 이를 적용한 DRM 시스템의 컨텐츠 암호화 모듈을 설계한다.

논문의 구성은 1절 서론에 이어서 2절 관련연구, 3절 적응적 보안 등급 결정 알고리즘, 4절 적응적 보안 등급을 적용한 DRM 시스템 설계 및 구현, 5절 사례연구, 6절 결론과 향후 연구에 대해서 논한다.

2. 관련 연구

2.1 DRM(Digital Rights Management)

DRM[1,2]은 컨텐츠의 지적 재산권이 디지털 방식에 의해서 안전하게 보유/유지 되도록 하는 시스템으로 정의할 수 있다. 즉, 안전한 저작권과 승인내역, 권리와 승인의 집행, 인증된 환경과 서비스 인프라등을 가능하게 하는 하드웨어와 소프트웨어를 모두 포함한 디지털 컨텐츠의 저작권 관리기술, 절차, 처리, 알고리즘 등을 포함한다. DRM을 이용한 디지털 컨텐츠 유통 시스템이 갖추어야 할 기본적인 요건으로 컨텐츠 보호/인증, Usage/Business Rule 적용, 라이선스 백

업, 클리어링하우스, Super-Distribution, 휴대용 단말기 지원 등이 있다. 현 DRM에서는 보안 정책 처리에 직접적인 영향을 미치는 사용자 디바이스의 성능이나 공격대상이 될 컨텐츠의 가치를 고려하지 않고 일률적인 보안 정책을 적용함으로 보안 정책이 효율적으로 적용되지 못하고 서비스 이용자에게 불편함을 주고 있다.

2.2 MAUT(Multi-Attribute Utility Theory)

MAUT[5](Multi-Attribute Utility Theory)는 다중변수에 대한 의사결정 문제(decision problem)에서 유틸리티(utility)를 통한 정략적인 의사결정 방법이다.

유틸리티 분석(utility analysis)은 의사 결정자(decision maker)가 원하는 제비뽑기(lottery)의 결과를 분석 해주는 분야로서 의사 결정자는 이들 결과에 대한 개인의 선호도(preference)를 유틸리티 수(utility number)로 표현한다.

유틸리티는 0과 1사이의 상대적인 값으로서 $u(x^0)$, $u(x^*)$를 각각 가장 선호하지 않는 결과 유틸리티와 가장 선호하는 결과 유틸리티라고 두면 $u(x^0)=0$, $u(x^*)=1$로 나타낸다. 그리고 결과에 대한 유틸리티 수의 대입은 기대 유틸리티(expected utility)를 최대화 시켜주는 쪽으로 이루어지고 기대 유틸리티의 최대화는 의사 결정자의 최적 행동의 기준이 된다.

MAUT는 다양한 곳에서 사용될 수 있으며 특히 Pmart(Pukyoung-mart)[6]는 가격 이외에 상품의 특성, 보장 기간, 서비스 정책 등 다른 조건에 대해 MAUT를 이용하여 협상할 수 있는 에이전트 중재에 의한 전자 상거래 프레임워크이다.

3. 적응적 보안 등급 결정 알고리즘

3.1 알고리즘을 위한 계수.

알고리즘을 위하여 계수를 정의한다.

① 디바이스의 성능에 대한 계수

device (0.5)	k_1	dNet	네트워크 속도
	k_2	dCpu	디바이스의 CPU 성능
	k_3	dRam	디바이스의 RAM 성능

② 디지털 컨텐츠 가치에 대한 계수

content (0.5)	k_4	cQuality	디지털 컨텐츠의 품질
	k_5	cExpire	디지털 컨텐츠의 사용기간
	k_6	cValue	디지털 컨텐츠의 가격

디바이스의 성능과 컨텐츠의 가치를 동일하게 고려하기 위해 전체 가중치를 각각 .5로 결정한다.

3.2 디바이스 성능 및 컨텐츠 가치의 환산 값.

다음은 각 변수의 최선 값(x^*_i)과 최악 값(x^o_i)을 토대로 0과 1사이의 값으로 변환된 값이다.

Attribute	x^o_1	x_2									x^*_1
dNet (kbps)	0	120	240	360	480	600	720	840	960	1180	1200
dCpu (Mhz)	0	300	600	900	1200	1500	1800	2100	2400	2700	3000
dRam (M)	0	160	320	480	640	800	960	1120	1280	1440	1600
환산된 값	0.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	1.0

cQuality(등급)	0	1	2	3	4	5	6	7	8	9	10
cExpire(일)	0	1	2	4	8	16	32	64	128	256	∞
cValue(원)	0	500	1000	1500	2000	2500	3000	3500	4000	4500	5000
환산된 값	0.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	1.0

3.3 보안 등급 결정 알고리즘

다음은 디지털 컨텐츠에 적용할 보안 등급을 결정하는 알고리즘이다.

```
function SecureModule returns SecuLevel
static
    // 디바이스의 성능
    네트워크 속도(dNet)  // 서비스에 직접적인 영향
    CPU 성능(dCpu)       // 보안정책처리에 직접적인 영향
    RAM의 성능(dRam)     // 보안정책처리에 직접적인 영향

    // 디지털 컨텐츠의 가치
    품질(cQuality)          // 컨텐츠의 질
    사용기간(cExpire)       // 사용기간에 관한 rule
    가격(cValue)            // 컨텐츠의 가격

    step 1 // 디바이스 성능 요구
      if sysNet() = SL0 {// 적용 범위 확인
        service deny;
      }
    step 2
        MAUT(); //MAUT에 따라 보안 등급 결정
  return SecuLevel ;
```

```
function SysNet() returns SecuLevel // 적용범위 확인
  If (dCPU = .1 || dRAM = .1 || dNet = .1) {
    SecuLevel = SL0; } //보안을 적용할 수 없는 등급
  return SecuLevel ;
```

```
function MAUT() returns SecuLevel
  static:
   u(x1,x2,..,xn)      : 유틸리티 함수
   k1,k2,..,kn         : 가중치, 선호도
   u(x1,..,xn) ← k1u1(x1)+...+knun(xn)
  //모든 i 에 대해 ui(x0i)=0, ui(x*i)=1, ki는 상수
  //u(xi), kn은 사용자와의 통신에 의해서 결정.
  for i=0 to n
    if risk prone then blog2(x+1)
    else if risk neutral then bx
    else if risk averse then b(2^(cx)-1), b, c>0
  end
 // 보안 등급 결정
  switch(u(x1,x2,..,xn))
     case(< 0.2) : SecuLevel=SL1;
     case(< 0.4) : SecuLevel=SL2;
     case(< 0.6) : SecuLevel=SL3;
     case(< 0.8) : SecuLevel=SL4;
     default SecuLevel=SL5;
return SecuLevel
```

4. 적응적 보안 등급을 적용한 DRM 시스템

본 논문에서는 에이전트를 이용하여 컨텐츠의 관리 및 사용자 디바이스 정보의 수집, License 관리를 수행한다. 또한 에이전트는 TRS[7]기능을 지원한다. 모든 데이터는 XML을 이용하여 표현함으로써 애플리케이션의 상호독립성을 확보하였고, License는 권리명세 언어인 XrML[8]을 이용하여 표현한다.

그림 1은 제안하는 DRM 시스템 구조이다.

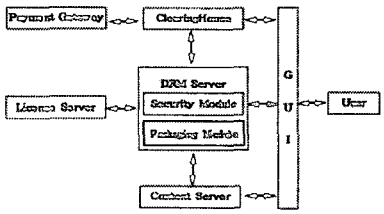


그림 1. Adaptive Security DRM Server

4.1 구성 요소

시스템의 각 구성요소의 기능은 다음과 같다.

- User : 컨텐츠 구매자로서 지불 및 권한을 가짐.
- License Server : 라이센스 관리 및 백업.
- Content Server : 컨텐츠 관리 데이터베이스.
- ClearingHouse : 컨텐츠 거래 내역 저장 및 보관, 결제 처리.
- DRM Server :
  + Security Module : 디바이스 정보와 사용자 선택 정보를 이용하여 보안 등급 결정.
  + Packaging Module : 결정된 보안 등급에 따라 컨텐츠에 보안 정책을 적용.

4.2 구현 및 평가

① 구현 환경

- 운영 체제 : Window 2000 Pro
- 사용 툴 : Mysql 3.23.47, Jakarta-Tomcat-4.0.3
- 개발 언어 : JDK 1.4.1-JAVA

② Security Module과 Packaging Module

본 논문에서 제안한 시스템에서 컨텐츠의 가치와 디바이스의 성능에 따라 MAUT 알고리즘을 이용해서 보안등급을 결정하는 Security Module과 결정된 등급에 따라 보안정책을 적용하는 Packaging Module은 현재 구현되어 있다.

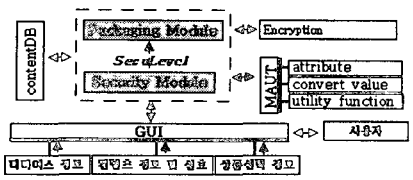
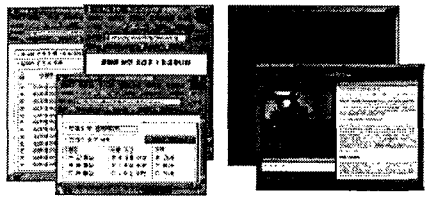


그림 2. Security · Packaging Module의 구성

Security Module은 그림 2와 같이 GUI(사용자와 통신), contentDB(컨텐츠 정보), MAUT(유틸리티 계산)등으로 구성되어 있다. 특히 사용자의 선호에 따른 유틸리티 함수의 종류와 디바이스의 성능에 따른 가중치는 미리 정의되었다. 이러한 정보들은 그림 2에서 사용자가 '컨텐츠 정보 및 선호'와 '상품 선택 정보'를 선택해 줌으로써 5 단계로 구분되는 보안등급이 결정된다. 이 보안 등급은 디바이스의 성능과 컨텐츠의 가치에 따라 다양한 등급으로 동적으로 결정된다.

Packaging Module은 결정된 보안 등급에 따라 차별적으로 컨텐츠를 암호화한다. 암호화된 컨텐츠는 사용자에게 전달되고 복호화 되어 서비스가 제공된다.



(a) (b)

그림 3. 실행화면

그림 3은 사용자가 원하는 상품 정보와 그에 따른 상품 목록을 선택 조건으로 제시하는 실행 화면(a)과 Packaging Module에서 컨텐츠를 암호화하는 단계와 디지털 컨텐츠 서비스를 제공받는 실행 화면(b)이다.

③ 평가

본 논문에서 제시한 Security Module은 디바이스 성능과 컨텐츠의 가치에 따라 동적으로 보안 등급을 결정하고 Packaging Module은 결정된 등급에 따라 차별적으로 보안 정책을 적용한다.

보안 정책이 디바이스의 성능과 컨텐츠의 가치에 따라 적절하게 결정되어 적용되므로 DRM 서버는 보안 정책을 적용시킬 때 context(디바이스 성능과 컨텐츠 가치 및 사용자 선호도)정보에 따라 적절한 보안 정책을 제시 할 수 있다.

5. 사례연구

사용자는 DRM Server에 접속하여 에이전트를 다운로드받는다. 에이전트는 사용자가 요구하는 컨텐츠와 디바이스의 정보를 DRM Server에게 제공한다.

Security Module은 사용자의 디바이스 성능을 평가하고, 최소 레벨 (SL1)을 만족하지 못하면 서비스를 거부한다. 사용자의 선호에 따라 $u_i(x_i)$를 결정한다. 디바이스의 성능에 따라 디바이스의 가중치 $\{k_1, k_2, k_3\}$를 동일하게 결정한다. 그리고 컨텐츠 가치의 가중치

표 1. 제시되는 컨텐츠 목록

	유틸리티(x1)	사용자 선호(x2)	가격(x3)
1	고 음질(.33)	∞ (.33)	5000원(.33)
2	고 음질(.42)	256 bit(.29)	4800원(.29)
3	고 음질(.62)	128 bit(.19)	4600원(.19)
4	중 음질(.26)	∞ (.37)	4700원(.37)
5	중 음질(.33)	256 bit(.33)	4500원(.33)
6	중 음질(.54)	128 bit(.23)	4300원(.23)
7	저 음질(.14)	∞ (.43)	4400원(.43)
8	저 음질(.18)	256 bit(.41)	4200원(.41)
9	저 음질(.33)	128 bit(.33)	4000원(.33)

$\{k_4, k_5, k_6\}$를 결정하기 위해서 표 1과 같이 사용자의 선호도를 질문한다. 가령 사용자가 표 1의 선택조건에서 3을 선택하였다면, 사용기간과 가격을 비 선호하고 품질만 선호하는 의미이므로 $k_4>k_5+k_6$, 즉, $k_4>.5$가 된다. 이와 같이 미리 정의된 표 1에 따라 $k_4=.62$, $k_5=.19$, $k_6=.19$로 결정된다. 그리고 만약 사용자의 선호에 따라 결정된 유틸리티 함수가 위험 회피형이라면 각 $u_i(x_i)$는 $\boldsymbol{blog_2}(x_i+1)$의 (concave)형태가 된다. 결과적으로 디바이스의 성능과 컨텐츠의 가치를 고려한 유틸리티 함수는 다음과 같다.

$u(x_1,x_2,x_3,x_4,x_5,x_6) = .5u(\text{디바이스 성능}) + .5(\text{컨텐츠 가치})$

$= .5\{k_1u(dNet)+k_2u(dCpu)+k_3u(dRam)\}$

$+ .5\{k_4u(cQuality)+k_5u(cExpire)+k_6u(cValue)\}$

$= .5\{.167\boldsymbol{log_2}(x_1+1)+.167\boldsymbol{log_2}(x_2+1)+.166\boldsymbol{log_2}(x_3+1)\}$

$+ .5\{.62\boldsymbol{log_2}(x_4+1)+.19\boldsymbol{log_2}(x_5+1)+.19\boldsymbol{log_2}(x_6+1)\}$

이와 같이 Security Module은 사용자의 선택에 따라 각 변수에 대한 k_i와 $u(x_i)$를 결정하고 MAUT를 이용하여 최적의 보안 등급을 결정하고 Packaging Module은 결정된 보안 등급에 따라 Content Server로부터 가져온 컨텐츠에 보안 정책을 적용함으로써 컨텐츠의 가치와 디바이스의 성능에 따라 차별적인 보안 정책이 적용될 수 있다. 이로써 보안 정책을 처리하는 디바이스의 성능과 컨텐츠의 가치에 적절한 보안 정책을 적용하게 된다.

이때 컨텐츠를 복호화하기 위한 정보는 License Server에 저장된다. Packaging된 컨텐츠는 사용자에게 전송되어 에이전트가 이를 관리한다.

컨텐츠가 다운로드 된 후 에이전트는 사용자에게 결제를 요청하고 사용자는 결제를 수락한다. 에이전트는 GUI를 통해 ClearingHouse와 결제 수단을 결정하고 결제를 수행한다. 거래내역은 ClearingHouse내에 저장되며 결제 여부를 License Server에게 전송한다. License Server는 해당 사용자의 License를 발급하여 에이전트에게 전달한다.

6. 결론 및 향후 연구

현재 DRM에서는 동일한 키 길이의 암호 알고리즘을 보안 정책으로 사용하고 있고, 보안 정책으로 인해 이용자들에게 보안 정책을 처리하기 위한 일정양의 자원과 시간을 요구한다. 이로 인해 각기 디바이스 성능이 다른 사용자들에게는 상대적으로 불편할 뿐만 아니라 보안 정책이 비효율적으로 적용되는 문제점을 가지고 있다.

따라서 본 논문에서는 이런 문제점들을 해결하기 위해 디바이스의 성능 및 컨텐츠의 가치를 MAUT에 적용하여 컨텐츠에 적용할 보안 정책을 동적으로 결정하는 알고리즘을 제안하였고, 이를 이용한 컨텐츠 암호화 모듈을 구현하였다. 이로써 다양한 보안 환경을 필요로 하게 될 인터넷영화관, 인터넷강의 등 디지털 컨텐츠 서비스에 본 논문에서 제시한 적응적 보안 등급 결정 알고리즘을 적용해 사용자들에게 편리성을 제공하고 보안정책을 효율적으로 적용할 수 있다.

향후 연구 과제로서 좀 더 다양한 변수들을 이용하여 보다 효율적으로 보안 정책을 적용하고 다양한 사용자의 요구사항을 반영할 수 있는 MAUT 알고리즘에 대한 연구가 필요하다.

참고문헌

[1] Qiong Liu and Reihan Safavi-Naini, "Digital Rights Management for Content Distribution," http://citeseer.nj.nec.com/560657.html .

[2] Joan Feigenbaum and Michael J. Freedman, "Privacy Enginnering for Digital Rights Management System," http://citeseer.nj.nec.com/feigenbaum01privacy.html .

[3] http://www.microsoft.com/windows/windowsmedia/drm.aspx .

[4] http://www.intertrust.com .

[5] R.L.Keeney and H.Raiffa, Decisions with Multiple Objectives: Preferences and Value Tradeoffs, John Wiley & Sons, New York, NY, 1976.

[6] Mokdong Chung and Vasant Honavar, "A Negotiation Model in Agent-mediated Electronic Commerce," Proc of the IEEE International Symposium on Multimedia Software Engineering, Taipei, Dec. 2000, pp. 403-410.

[7] D. Aucsmith and Graunk, "Tamper Resistant Software: An Implementation," Proc of the 1st International Workshop on Information Hiding, Springer Lecture Notes, 1986.

[8] XrML "eXtensible rights Markup Language" http://www.xrml.org/ .