침입감내기술에서의 Voting 및 그룹관리 신뢰성 분석

이태진*, 김형종*,이강신*

Analysis of the Dependability of Voting and Group Management in the Intrusion Tolerant Technology

Tai Jin Lee*, Hyung Jong Kim* and KangShin Lee*

Abstract

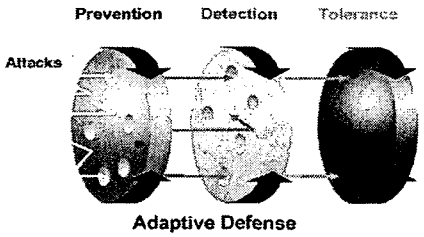
Intrusion tolerant technology is the technology to guarantee the quality of service for certain amount time from the attacks which cannot be defended by the previous information security technologies. It increases the availability and confidentiality of the system by minimizing the damage from the attacks. And the fundamental components of the intrusion tolerant technology are voting and GMP(Group Management Protocol). In this paper, we present a new scheme to analyze the voting dependability and corrupt member detection dependability, which is very critical in GMP. Based on this scheme, we can make a new security policy and the methodology of analyzing the dependability itself also can be applicable to the other field.

Key Words: GMP, GCS, Voting, Intrusion Tolerance System,

* 한국정보보호진흥원 기반보호기술팀
** 저자 2의 소속

1. 서론

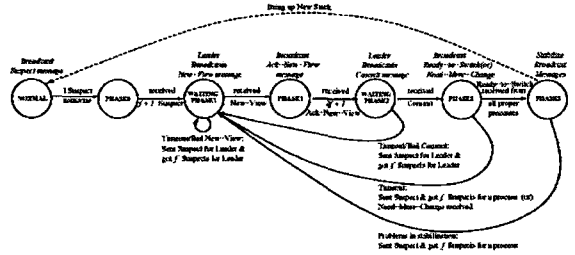
최근 발견되는 소프트웨어의 취약점이 연간 2000여건에 이르며, 컴퓨터 보안사고 건수도 급증하여 정보통신시스템에 대한 위협이 증가하고 있다. 침입감내기술은 이러한 서비스거부 공격, 악의적인 코드, 내부 위협등의 공격이 발생하더라도 저항, 인식, 적응, 복구로 정의할 수 있다.



<그림 1. 침입감내 개념>

이러한 침입감내기술은 기존의 침입예방기술, 탐지기술로는 대응할 수 없는 알려지지 않는 공격으로 인한 피해를 최소화시켜 서비스의 가용성과 신뢰성을 향상시킨다. 침입감내기술 중 가장 기본적인 요소는 같은 request에 대해 중복된 서버가 이를 수행하고 다수의 서버가 같은 response를 산출하면 이를 클라이언트에게 전달하는 Voting과 이에 참여하는 서버들을 유기적으로 관리하는 그룹관리이다.

침입감내기술에 대한 연구는 ITUA, ITDOS, MAFTIA, Immune같은 프로젝트를 통해 수행되었는데, 이들 역시 복제된 서버를 이용한다. 대표적인 프로젝트인 ITUA에서는 아래그림과



<그림 2. ITUA 손상된 멤버탐지 및 재구성 과정>

같은 메커니즘으로 손상된 멤버를 탐지하고 그룹을 재구성한다.

ITUA에서는 위 그림과 같은 과정을 통해 손상된 멤버를 탐지하고 재구성하는 과정을 거친다. 멤버가 crash할 경우에 감내할 수 있는 C-Ensemble기반의 GCS에 프로토타입을 실제로 개발하고, 손상된 멤버를 실제로 주입하여 이 멤버가 탐지되고 그룹이 재구성되는 성능평가 작업을 하였다. 그러나 모든 멤버가 손상된 멤버를 탐지하고 잘못된 탐지를 하지 않음을 가정하므로, 성능중심의 분석이 이루어졌고 그 신뢰성은 검증되지 않았다.

여기서는 침입감내기술에서 Majority Voting의 Voting정확성, Voting의 실패확률 및 이를 이용하여 손상된 멤버를 탐지하는 탐지신뢰성을 분석한다. 이 분석을 통해 침입감내기술이 적용된 특정시스템의 보안수준을 정하는데 참고자료가 될 수 있으며, 신뢰성을 분석하는 방법 또한 가이드를 준다.

2. Majority Voting

2.1 Majority Voting 신뢰성 분석

Majority Voting은 복제된 서버가 같은 request를 수행하고, 그 결과 중 다수의 서버에 의해 같은 response가 산출되면 그것을 옳은 response를 간주하는 Voting방식이다.

Corrupt : 손상되어 request에 대해 정확한 response를 산출하지 못하는 멤버상태
Correct : request에 대해 정확한 response를 산출할 수 있는 멤버상태
n : Voting에 참여하는 멤버 수
p : 멤버가 Corrupt 될 확률
m : 옳은 결과로 간주하는 기준이 되는 수

- **Assumption**

Correct한 멤버는 정확한 response를 산출함
Corrupt된 멤버는 정확한 response를 산출하지 못함

- **P(condition, result, number)**
condition : 정확한 response와 Voting에서 산출된 response의 동일여부 (Correct/ Incorrect)
result : Voting에 의해 산출된 결과 (Select/ Not Select)
m : Voting에서 정확한 response를 산출한 것으로 인정하는 멤버 수(Number)

P(Correct, Select, m) + P(Incorrect, Select, m) + P(Incorrect, Not Select, m) + P(Correct, Not Select, m) = 1

P(Incorrect, Not Select, m)은 Voting을 통해 정확한 response를 산출하지 못한 것으로 간주하므로 문제가 있음을 알 수 있게 되어 Voting의 신뢰성에 영향을 주지 않는다. 그러나 P(Correct, Select, m)는 정확한 Voting결과를 산출한 것이고, P(Incorrect, Select, m)은 잘못된 response를 Voting에서 정확한 결정으로 간주하게 되므로, Voting의 신뢰성을 측정하는 기준이 된다.

2.2 Majority Voting 신뢰성 계산

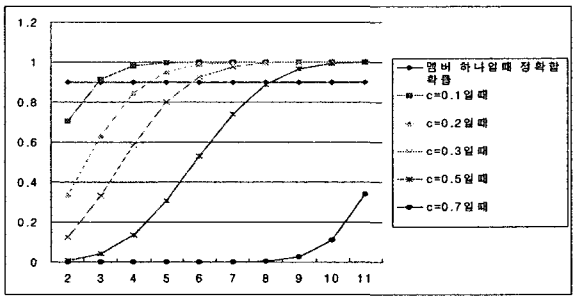
m개 이상의 멤버가 Correct하면, 정확한 결과를 낼 수 있으므로

$$P(\text{Correct, Select, m}) = \sum_{k=m}^{n} {}_nC_k \, (1-p)^k \, p^{n-k}$$

Corrupt한 멤버는 request에 대해 임의의 response를 산출한다. 여기서 Corrupt한 멤버가 같은 response를 산출하는 값들에 대해 그룹을 형성하며, response 수에 따라 group1, group2와 같이 일련번호를 붙일 수 있다. 여기서 group1의 response가 나올 확률을 c라 할 때, c에 따라 P(Correct, Not Select, m)을 계산할 수 있다.

$$P(\text{Incorrect, Select, m}) = \sum_{k=m}^{n} [\, {}_nC_k \, (1-p)^{n-k} \, p^k \times (\sum_{s=m}^{k} {}_kC_s \, c^s \, (1-c)^{k-s} \,) \,]$$

2.3 결과 분석



<그림 3. n=12,p=0.1일때, c에 따른 Voting신뢰성>

c는 복제된 서버들이 같은 공격에 대해 동시에 어느 정도 Corrupt되는지를 나타내는 지수가 된다. 같은 공격에 의해 복제된 서버가 동시에 피해를 입지 않는 경우엔 c가 작아지고 Voting의 신뢰성은 높아진다. 그러나 같은 공격에 의해 복제된 서버가 동시에 상당한 피해를 입게 될 때 c는 큰값을 가지게 되며, Voting의 신뢰성은 크게 떨어지게 되어 복제된 서버를 유지할 필요가 없어지게 된다. 위의 그림을 보면 c가 0.5정도의 값을 가지면 Voting을 통해 신뢰성 향상을 기대하기 어렵다.

3. Voting기준에 따른 탐지 신뢰성

3.1 개별적인 그룹탐지 분석

$D_{corrupt}$: 임의의 멤버A가 임의의 멤버B를 Corrupt된 것으로 탐지할 확률
$D_{correct}$: 임의의 멤버A가 임의의 멤버B를 Correct한 것으로 탐지할 확률

그룹관리에서 개개의 멤버가 다른 멤버의 Corrupt여부를 탐지할 때
r : Corrupt된 멤버를 정확히 탐지할 확률
s : Correct한 멤버를 Corrupt로 잘못 탐지할 확률

그림으로 도식화하면 다음과 같다.

		탐지대상 실제상태	
		Correct	Corrupt
탐지주체 실제상태	Correct	Correct로 판단 (D4) [정상결과]	Corrupt로 판단 (D1) [정상결과]
		Corrupt로 판단 (D3) [오판결과]	Correct로 판단(D2) [오판결과]
	Corrupt	Correct로 판단 (D6) [정상결과]	Correct로 판단 (D8) [오판결과]

<표 1. 탐지대상의 실제상태와 탐지결과와의 관계>

$D_{corrupt}$ = D1 + D3

$D_{correct}$ = D2 + D4 + D6 + D8

$D_{corrupt}$ + $D_{correct}$ = 1

Corrupt로 탐지했을때, 그 판단이 정확할 확률 = $\frac{D1}{D1 + D3}$

Corrupt로 탐지했을때, 그 판단이 틀릴 확률 = $\frac{D3}{D1 + D3}$

Correct로 탐지했을때, 그 판단이 정확할 확률 = $\frac{D4 + D6}{D2 + D4 + D6 + D8}$

Correct로 탐지했을때, 그 판단이 틀릴 확률 = $\frac{D2 + D8}{D2 + D4 + D6 + D8}$

3.2 전체적인 그룹탐지 분석

임의의 멤버A에 대해 n-1개의 멤버가 탐지를 하고, 4개의 구분으로 나눌 수 있다.

		멤버 A에 대한 그룹 탐지결과	
		Corrupt (m개이상 Detect)	Correct (m개미만 Detect)
멤버A 실제상태	Corrupt	(P1) m개이상 멤버 Detect, 멤버 A Corrupt	(P2) m개미만 멤버 Detect, 멤버 A Corrupt
	Correct	(P3) m개이상 멤버 Detect, 멤버 A Correct	(P4) m개미만 멤버 Detect, 멤버 A Correct

<표 2. 그룹탐지결과 분류>

P1 + P2 + P3 + P4 = 1

$P1 + P3 = \sum_{k=m}^{n-1} {}_{n-1}C_k \; D_{corrupt}{}^{k} \; D_{correct}{}^{n-1-k}$

P(t) : Corrupt된 임의의 멤버 A에 대해 t개의 멤버가 Corrupt된 것으로 탐지할 확률

P(t, m) : P(t)에서 정확히 탐지한 멤버수가 m개 이상일 확률

$P(t) = {}_{n-1}C_t \; D_{corrupt}{}^{t} \; D_{correct}{}^{n-1-t}$

- 0 <= i < m , n - 1 - t < m - i 일때

이 경우, 정확히 Corrupt로 탐지할 멤버수가 m개가 되지 않는다.

P(t, m) = 0

- 0 <= i < m , n - 1 - t >= m - i 일때

$$P(t, m) = \sum_{i=0}^{t} {}_tC_i \left(\frac{D1}{D1 + D3}\right)^i \left(\frac{D3}{D1 + D3}\right)^{t-i} \left[\sum_{j=m-i}^{n-1-t} {}_{n-1-t}C_j \left(\frac{D2 + D8}{D2 + D4 + D6 + D8}\right)^j \left(\frac{D4 + D6}{D2 + D4 + D6 + D8}\right)^{n-1-t-j}\right]\right]$$

- m < i 일때

$$P(t, m) = \sum_{i=0}^{t} {}_tC_i \left(\frac{D1}{D1 + D3}\right)^i \left(\frac{D3}{D1 + D3}\right)^{t-i}$$

$$P1 = \sum_{t=m}^{n-1} P(t)\, P(t, m)$$

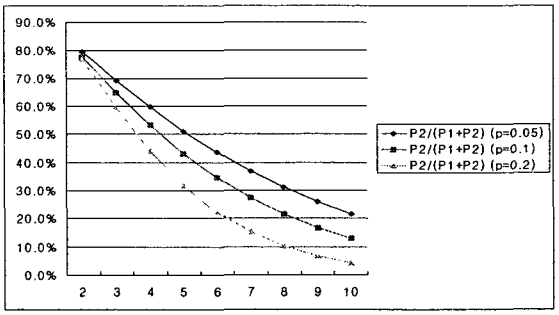
P2도 같은 방식으로 구할 수 있다.

특정 멤버A를 Corrupt된 것으로 탐지했을때

그룹탐지 정확률 = $\frac{P1}{P1 + P3}$

그룹탐지 오탐지율 = $\frac{P2}{P1 + P2}$

3.3 결과 분석



<그림 4. p에 따른 그룹탐지 정확률 >

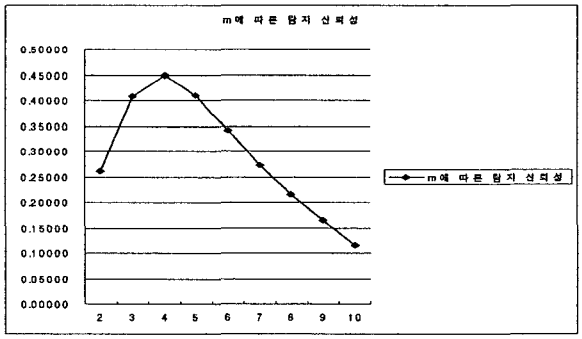
m이 커짐에 따라 Corrupt로 탐지하는 기준이 엄격해지므로, Corrupt된 멤버로 탐지할 확률이 작아지고, p가 작을수록 탐지율도 떨어지게 된다.

3.4 그룹탐지 신뢰성 종합

앞에서 산출한 그룹탐지 정확성은 특정멤버 A가 Corrupt된 것으로 m개 이상의 멤버가 동의하면 A는 Corrupt된 것으로 간주할 때 정확히 탐지할 확률을 계산한 것이다. 따라서 그룹탐지의 신뢰성은 Voting의 신뢰성을 고려해야 한다.

그룹탐지 신뢰성 = Voting 기준에 대한 신뢰성 × Voting기준에 따른 탐지 신뢰성

이를 도표로 나타내면 다음과 같다.



<그림 5. 그룹탐지 신뢰성도표>

즉, n=12, p=0.1, c=0.2일때 그룹탐지의 신뢰성을 가장 높일수 있는 m은 4라는 결과를 산출했다. 침입감내기술이 적용된 특정 시스템에서 p, c는 정해지게 되며 침입감내를 위한 m을 설정할 때 이와 같은 분석을 통해 신뢰할 수 있는 최적의 보안정책을 운영할 수 있다. 또한 시스템을 구성할 때 정확한 그룹탐지 확률, Corrupt된 멤버를 탐지못할 확률, 오탐지율등을 통해 보안수준을 유지하기 위한 다른 차선책들을 고려하는 기준이 된다.

4. 결론

본 논문에서는 침입감내시스템의 핵심요소인 Voting과 그룹탐지의 신뢰성분석방법에 대해 이론적인 확률을 통해 제안하였다. Voting과 그룹탐지에서 기본적인 factor외에 각각의 멤버가 손상될 확률, Voting실패에서의 확률, 멤버탐지에서 정확히 탐지할 확률, 오판할 확률의 요소를 두어 침입감내기술이 적용된 특정 시스템의 Voting 및 그룹탐지의 신뢰성을 분석할 수 있다.

신뢰성 분석을 통해 시스템 보안정책을 수립할 때 참고자료로 활용가능하며, 시스템 상황에 맞는 적절한 Voting, 그룹탐지 수준을 정할 수 있다. 또한 여기서의 신뢰성분석 방법은 신뢰할 수 없는 노드들간의 신뢰도를 측정하는 하나의 방법이 된다.

본 논문에서의 신뢰성 분석방법은 formal한 검증없이 수식으로 결론을 이끌어 냈다. 향후에 이를 증명할 수 있는 모델 및 시뮬레이션을 통해 증명을 해야하며, 이 모델을 추상화하여 적용분야를 다양화 할 수 있을 것이다.

참고문헌

[1] James Patrick Lyons, "A Replication Protocol for An Intrusion-Tolerant System Design", University of Pennsylvania, 2000.
[2] Harigovind Venkatraj Ramasamy B.engr,

"A Group Membership Protocol for An Intrusion-Tolerant Group Communication System", Anna University, 1999.
[3] Adnan Agbaria, Roy Friedman, "Overcomming Byzantine Failures Using Checkpointing"
[4] G. Bracha and S. Toueg, "Asynchronous Consensus and Broadcast Protocols", Journal of the ACM
[5] Prashant Pandey, B.ENGR, "Reliable Delivery and Ordering Mechanisms for an Intrusion-Tolerant Group Communication System", Birla Institute of Technology and Science, 1999.
[6] Michael K. Reiter,"Reliable and Atomic Group Multicast in Rampart",AT&T Bell Laboratories, Holmdel, New Jersey, U.S.A
[7] Harigovind V, Ramasamy, Michel Cukier, "Formal Verification of an Intrusion-Tolerant Group Membership Protocol", IEICE TRANS, December 2003, No 12, Vol. E86-D