情 報 保 護 學 會 誌
第 13 券 第 1 號, 2003. 2

침입감내기술 연구 동향

최 중 섭*, 이 경 구*, 김 홍 근*

요 약

침입감내기술은 기존의 침입차단이나 탐지기술에 의하여 해결될 수 없었던 알려지지 않은 취약점을 이용하는 공격에 의하여 발생하는 시스템의 피해를 방지하기 위한 기술이며, 중요 서비스의 품질요구사항과 지속성 요구사항 만족을 위하여 의존성 특성의 만족이 필요하다. 이러한 침입감내시스템은 결함허용기술과 정보보호기술이 결합된 형태로 접근이 시도되고 있다. 침입감내기술에서는 일반적으로 결함허용기법들이 고려하고 있는 우발적 사고가 아닌 악의적 공격에서 일어날 수 있는 상황들에 대한 고려가 매우 중요하다. 그러므로 침입감내기술에서는 결함허용기술에서 고려하는 것 외에 보안취약성과 공격 개념의 도입, 침입의 탐지와 대처 등 보안성에 대한 요구사항 만족이 필요하다.

Ⅰ. 서 론

최근, 정보통신기반이 급속히 발달하고 사용자가 늘어남에 따라 이와 관련된 여러 가지 역기능들이 발생하고 있다. 최근에 보고되는 침해사고들은 점점 더 복잡하고 교묘한 공격방법들을 사용하여 피해 규모를 키워가고 있다. 또한 침해사고의 원인이 되는 시스템, 네트워크, 소프트웨어의 취약성들이 년간 3000여개 이상 발견되는 등 잠재적인 침해사고의 원인들도 매우 급속하게 증가하고 있다.

이러한 침해사고를 예방하고 효과적인 대응방법을 마련하기 위하여 침입차단기술, 침입탐지기술 등의 다양한 정보보호기술들이 개발되고 있다. 그러나 이런 기술들은 알려진 취약점에 대한 공격 예방과 탐지를 제공하며, 알려지지 않은 취약점에 대한 공격에는 적절한 대응이 쉽지 않은 단점이 존재한다. 그러므로 이와 같이 알려지지 않은 취약점이나 공격 방법에 의한 침해사고를 방지하기 위한 기술이 필요하며 침입감내기술 (Intrusion Tolerance Technology)이 이에 대한 한 가지 해결책으로 제시될 수 있다.

침입감내기술이란 중요한 서비스를 제공하는 시스템에 대한 공격이 발생하더라도 정상적인 서비스를 제공할 수 있도록 하는 기술이다.

이러한 침입감내기술은 결함허용기술과 침입차단기술이나 탐지기술 등의 정보보호기술들이 결합된 형태로 미국과 유럽에서 비교적 최근에 시작되었다. 침입감내 기능을 제공하기 위해서는 결함허용을 위한 기법들 외에 우발적 사고가 아닌 악의적 공격에서 일어날 수 있는 상황들에 대한 고려가 추가적으로 필요하다. 악의적 공격에 대처하기 위해 추가적으로 고려하여야 하는 것은 취약점과 공격 개념의 도입, 침입의 탐지와 대처, 피해의 복구, 원인의 제거 등과 같은 분야가 있다. 침입감내기술에 대한 연구는 오래전부터 수행되어 왔으나 많은 연구자들이 집중적으로 연구를 시작한 것은 비교적 최근의 일이다. 유럽에서는 FP5의 IST 프로그램을 통하여 관련 연구를 진행하였고, 미국에서도 DARPA의 지원을 통하여 연구들을 진행하고 있다.

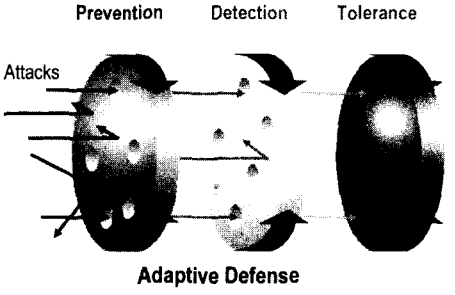
이 논문은 다음과 같이 구성된다. 이 논문의 2장에서는 계층적 정보보호의 개념과 의존성 개념에 대하여 설명함으로써 침입감내기술의 개념을 설명하고 감내하여야 하는 대상에 대하여 정의한다. 3장에서는 침입감내기술의 요소기술들을 식별하고 각각의 기능들을 구현할 때 고려하여야 하는 사항들에 대하여 논의한다. 4장에서는 미국과 유럽에서 진행되었던 관련 연구들을 분석한다.

본 연구는 정보통신부 선도기술개발사업 (능동형네트워크정보보호기술개발)의 지원을 받았음
* 한국정보보호진흥원 ({jschoi, kglee, hgkim}@kisa.or.kr)

II. 침입감내기술

1. 계층적 정보보호 개념

침입감내기술(ITS : Intrusion Tolerant System Technologies)은 취약성분석기술, 침입차단기술, 침입탐지기술과 같이 시스템에 대한 불법적인 침입이나 공격에 대한 대책 마련을 위하여 시도되는 새로운 정보보호기술이다.

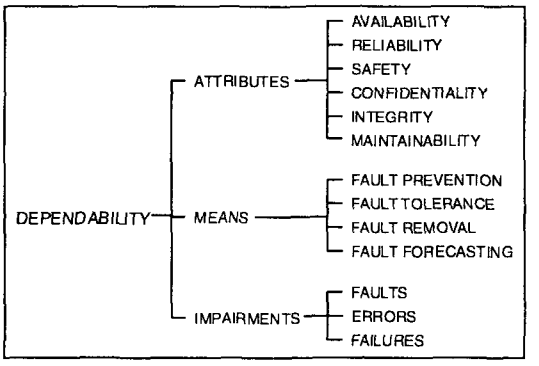


〔그림 1〕 계층적 정보보호의 개념

침입감내기술의 설명을 위해서는 먼저 그림 1에 표시된 계층적 정보보호 개념을 알아야 할 필요가 있다. 이 그림은 정보통신기반 보호를 위하여 정보보호기술들이 어떻게 적용되는가를 보여준다. 첫 번째 계층은 예방(Prevention) 계층이다. 이 계층은 시스템이 가지는 취약성 분석, 방화벽 기술 등과 같이 공격 예방을 위한 기술들을 표현한다. 다음 계층은 탐지 (Detection) 계층이다. 이 계층에서는 예방계층에서 미처 방어하지 못한 취약점을 뚫고 침입하는 공격을 탐지하고 이에 대한 대응책을 마련하는 기술들을 표현한다. 현재 대부분의 정보보호기술은 이 두 계층을 구현하고 있다. 그러나 이 계층들에서 대책 마련이 어려운 알려지지 않은 취약점은 얼마든지 존재가 가능하다. 감내 (Tolerance) 계층은 이와 같이 알려지지 않은 취약성으로 인한 침입에 대한 대책을 표현한다. 이 계층에서는 두 계층을 모두 뚫고 침입하는 고도의 기술을 가진 침입자들에 대한 대책을 마련하며 복제 시스템을 이용하는 침입감내기술이 대표적인 해결책이다.

2. 침입감내 개념

중요한 서비스가 정상적으로 유지되기 위한 조건으로 '가용성', '신뢰성', '안전성', 그리고 '보안성'이 유지되어야 한다. 이런 특성을 의존성 (dependability) 특성이라고 한다[1].

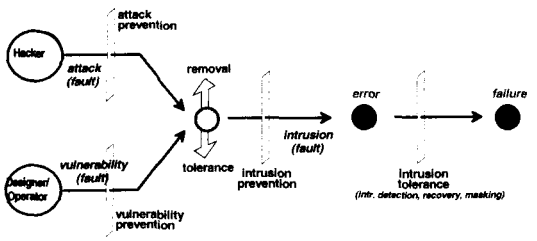


〔그림 2〕 의존성 특징

의존성 특성이 만족되지 않는다면 해당 시스템은 신뢰할 수 있는 상태라고 볼 수 없고, 의존성 특성을 해치는 기본적인 원인은 시스템의 손상이다. 시스템의 손상은 일반적으로 결함으로부터 시작되며, 이러한 결함은 우발적, 혹은 의도적으로 발생할 수 있고, 발생 요인 또한 내부적 결함 혹은 외부환경에 의한 결함, 설계의 결함 등 매우 다양한 원인으로부터 발생할 수 있다. 결함은 시스템의 다른 부분으로 전파되어 시스템이 정상적인 결과물을 산출하지 못하는 상태인 '오류' 상태로 전이되고 오류 상태는 결국 시스템을 '고장' 상태로 전이되게 된다. 결함허용 기술은 시스템의 결함이 '오류' 혹은 '고장' 상태로 전이되어 정상적인 서비스가 불가능하게 되는 것을 방지하기 위한 기술이다.

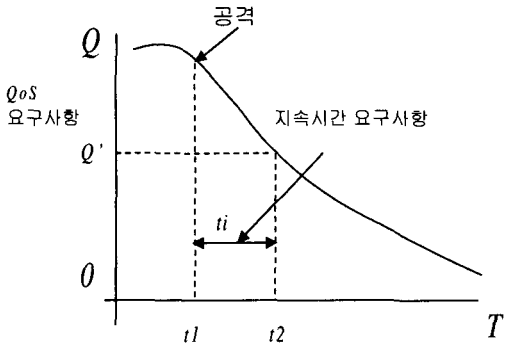
결함허용개념에서 관심을 두고 있는 결함의 종류는 시스템 설계의 오류, 하드웨어의 부분적 파손에 의한 작동 불능, 운영자의 실수에 의한 이상 결과 산출 등 주로 시스템 내부적으로 유발되거나 악의가 없는 우발적 실수에 의하여 발생하는 결함이 대부분이다. 그러나, 소프트웨어의 설계 결함을 악용하는 공격, 시스템의 취약점을 이용하는 공격 등의 악의적 공격에 의하여 발생하는 결함은 결함허용기법을 사용하더라도 효과적으로 대응될 수 없는 경우가 대부분이다.

침입감내기술은 중요한 서비스를 제공하는 시스템에 적용함으로써 악의적 공격이 발생하였을 경우에도 원래의 시스템이 제공하는 정상적인 서비스를 일정한 시간동안 지속적으로 제공하여 주기 위한 기술이다.



〔그림 3〕 Fault, Error, Failure, Intrusion 개념

침입감내를 위한 요구사항은 서비스의 품질수준 요구사항과 지속성 요구사항으로 나타낼 수 있고 이것은 그림 4와 같이 표현된다. 즉 Q의 수준을 유지하던 서비스에 공격이 발생하는 경우 공격에 대한 적절한 대응을 통해 서비스의 품질 저하를 방지하여 Q'이상의 품질을 가진 서비스를 시스템 관리자나 보안전문가가 문제를 인식하고 해결될 때까지의 시간인 지속시간 요구사항(ti) 이상으로 지속적으로 제공할 수 있어야 한다는 것이다.



〔그림 4〕 침입감내 요구사항

침입감내개념에서는 결함허용기술의 결함에 대한 정의에 보안결함을 추가적으로 고려하여야 한다. 보안결함은 바이러스나 웜 등의 악성코드와 시스템·네트워크에 대한 침입을 가능하게 하는 취약성이 있다. 표 2는 침입감내기술에서 취급하여야 하는 결함들을 구분하여 표시한 것이다.

〔표 1〕 결함허용 개념과 침입감내 개념의 비교

	결함허용 개념	침입감내 개념
관심대상	- 고장 : 시스템이 원래 의도된 기능을 만족하지 못하게 된 상태 - 오류 : 시스템의 부분이 의도된 기능을 발휘하지 못하게 된 상태 - 결함 : 오류가 일어난 원인에 대한 판정 또는 가정 즉, 결함 → 오류 → 고장	- 공격 : 악의적 행위 자체 - 취약점 : 시스템의 약점이나 결점 - 침입 : 외부적으로 유도된 의도적 혹은 악의적인 운영상 오류이며, 취약점과 공격이 침입의 원인 즉, (취약점+공격) → (침입,결함) → 오류 → 고장
결함/오류의 원인	- 설계의 실수 - 하드웨어 결함 - 외부환경	- 설계자의 악의적 코드 삽입 - 악의적 파손 - (내, 외부인의) 침투

〔표 2〕 결함의 구분

구분	Nature		원인						지속성	
			현상		위치		생성 단계			
	우발적 결함	의도적 결함	물리적 결함	인간이 만든 결함	내부 결함	외부 결함	설계 결함	운영상 결함	항구적	일시적
물리적 결함	O		O		O			O	O	
	O		O			O		O	O	
순간적 결함	O		O			O		O		O
간헐적 결함	O		O		O			O		O
	O			O	O		O			O
설계결함	O			O	O		O		O	
상호작용 결함	O			O		O		O		O
악성 코드		O		O	O		O		O	
		O		O	O		O			O
침입		O		O		O		O	O	
		O		O		O		O		O

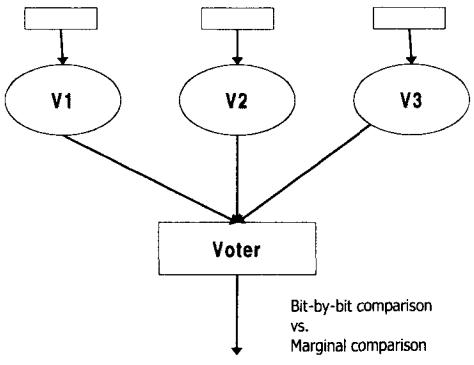
침입감내기술은 많은 부분에서 결함허용기술에서 얻어진 개념과 기술을 공유한다. 그러나 결함허용기술이 시스템과 소프트웨어의 우발적인 결함에 대하여 관심을 가진다면 침입감내기술은 이와 같은 범주의 결함 이외에 악의적인 행위에 의하여 발생하는 결함 즉 침투, 바이러스, 웜 등과 같은 악성코드나 침입에 대하여 관심을 가진다는 것이 다른 점이다.

III. 침입감내기술의 요소기술

1. 중복된 시스템 (Redundant systems)

의존성을 높일 수 있는 방법으로 서비스를 제공하는 시스템을 중복시키는 방법을 사용할 수 있다. 중복 시스템은 여러 측면에서 복제된 객체를 사용함으로써 시스템에 결함이 발생하였을 때 다른 복제본에서 얻어진 정상적인 결과를 사용할 수 있도록 하는 방법이다. 즉, 같은 기능을 하는 여러 수준의 복제본(replica)을 채용함으로써 일부 특정 기능의 서비스가 불가능하게 되었거나 잘못된 결과를 산출할 때 정상적인 것으로 대체할 수 있도록 있어야 한다.

복제의 대상은 하드웨어, 프로그램, 정보/설계, 데이터, 통신프로그램, 그리고 시간적 중복 등이 있다. 이와 같은 복제는 단지 같은 기능을 하는 동일 객체에 대한 동일 복사본을 제공하는 것 뿐 아니라 다른 플랫폼의 하드웨어 사용, 같은 기능의 소프트웨어를 다른 설계자들이 중복적으로 하는 이중 설계, 시간적 차이를 두고 결과를 산출하도록 하여 시스템이 갖는 간헐적 결함에 의하여 이상이 발생하지 않도록 하는 시간적 중복 등 한가지의 취약성이 복제 시스템에서도 발생하지 않도록 하는 대책이 내포되어야 한다.



〔그림 5〕 복제에 의한 결함 마스킹 예 (NVP)

그림 5는 소프트웨어의 복제를 통하여 결함을 방지하는 한가지 예를 보여준 것이다. 이 방법은 결함허용 시스템에서 많이 사용되고 있는 NVP (N-Version Programming)의 예이다. 이 방법은 설계나 구현 혹은 플래폼을 달리하는 N개의 복제본을 사용하여 각각의 결과를 산출한 다음 이 결과들을 비교하여 비정상적인 행위를 하는 버전을 검출함으로써 결함이 발생한 버전의 결과가 시스템에 전파되지 않도록 하는 예이다.

중복의 형태는 FRS (Fragmentation-Redundant-Scattered) 특징을 만족하는 것이 매우 중요한 이슈가 될 수 있다. 즉, 한 중복시키고자 하는 객체를 같은 복사본으로 저장하는 것보다 해당하는 객체를 조각내어 복사한 후 여러 위치에 저장함으로써 비밀성과 가용성을 향상시킬 수 있다. 그러나 이러한 중복 시스템은 비용의 증가를 초래한다. 침입감내기술이 적용되는 분야가 실질적으로 많은 응용에 적용되려면 중복성을 적용하는 연구에서 비용에 대한 타협점을 찾는 것도 중요하다.

2. Security 보장

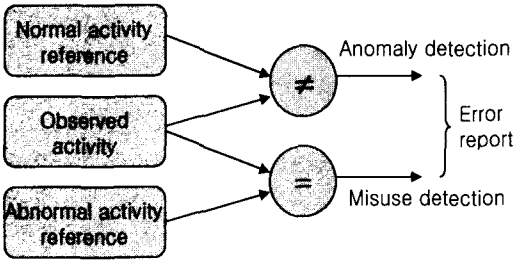
2.1 오류 처리

오류 처리는 결함 탐지와 복구에 의하여 이루어진다. 오류 탐지는 시스템의 결함이나 공격이 발생하였는지 알기 위한 방법이며 복구는 결함이나 공격이 발견되는 경우 시스템이 정상적인 상태를 유지할 수 있도록 하기 위한 방법이다.

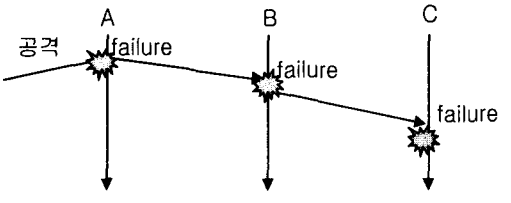
결함이나 침입 증상의 탐지는 수용검사나 침입탐지센서 등의 각종 탐지기술에 의하여 이루어질 수 있다. 침입탐지기술은 결함탐지보다 더 복잡한 고려가 필요하다. 즉 침입탐지 센서에 의하여 탐지되는 침입이 항상 정확한 결과가 아닐 수 있으므로 (즉, 침입자의 속임수나, False Positive, False Negative 등의 오탐지) 탐지의 정확성을 확보하기 위한 방법이 필요하다. 탐지의 정확성은 성능이 좋은 센서를 개발하거나 그림 6과 같이 여러 가지 탐지 방법을 혼용하거나 수용검사와 탐지 결과를 조율하는 방법을 사용함으로써 확보 가능하다.

결함이 탐지되면 오류가 발생한 부분을 시스템에서 분리하고 오류가 없는 새로운 부분으로 대치(복구)하여야 한다. 복구는 중복된 시스템이나 모듈을 준비함으로써 해결될 수 있다. 그러나 국지적인 하드웨어 결함의 경우에는 결함이 발생한 특정 부분을

제거하고 미리 준비된 백업시스템으로 서비스를 대체하는 방법으로 해결될 수 있지만 공격으로 인하여 발생하는 결함의 경우에는 이와 같은 방법이 효과적인 해결수단이 될 수 없는 경우가 많다.



〔그림 6〕 침입탐지기술



〔그림 7〕 연속적인 서비스 중단

예를 들어 복제 노드로 구성된 시스템이 서비스거부공격과 같은 원인에 의하여 한 노드가 정상적인 기능을 수행하지 못하고 다른 노드로 서비스를 이전하여 복구했더라도 공격 원인이 제거되지 않아서 공격이 계속된다면 그림 7에서 보여진 것과 같이 다른 노드들도 연속적으로 고장 상태로 전이될 가능성이 높다. 그러므로 주된 감내 대상이 공격인 침입감내 시스템에서는 고장 원인의 제거 혹은 고립화가 중요한 역할을 할 수 있다.

2.2 결함치료

결함이 발생한 부분을 시스템에서 분리하는 과정이 수행된 후에는 발생한 결함에 대한 치료가 필요하다. 이 부분에서는 어떠한 결함 혹은 침입이 발생하였고 어떤 오류를 발생하게 하였는지 식별하고 예방조치를 취하는 기능이 필요하고 오류 진단기능이 이 부분을 처리할 수 있다.

진단 기능은 시스템의 피해 정도 진단, 발생한 결함 식별, 원인 식별 등을 위하여 필요한 기능들이 제공되어야 한다. 이를 통해 일단 발생된 결함이나 피해를 복구하거나 결함이 다른 지역으로 전파되는 것을 방지하고, 결함을 야기한 취약점을 수정함으로써 같은 공격이 다시 발생하지 않도록 하여야 한다.

3. 결함 회피

결함 회피는 높은 품질의 설계와 프로그램을 가능하도록 하여 결함발생확률을 낮추는 방법이다. 설계에 대한 검증은 정형기법에 의하여 행해질 수 있다. 정형기법은 시스템을 수학적 혹은 기호적 표현으로 나타내고 표현된 문제의 정확성을 증명할 수 있는 도구를 제공함으로써 소프트웨어의 검증을 가능하게 해 준다. 소프트웨어의 정형적 분석을 위한 도구는 Z, RTL 등과 같은 수학적 증명을 통하여 소프트웨어의 정확성을 주장하는 것들과 스테이트차트와 같이 그래픽 표시에 의하여 설계를 표현하고 정확성을 검증할 수 있도록 도와 주는 것들이 있다. 이런 도구를 사용하는 것 외에 신뢰성이 높은 하드웨어 부품을 사용하고 숙련도가 높은 프로그래머에 의하여 구현된 소프트웨어 부품을 사용함으로써 결함을 회피하는 방법도 가능하다.

이러한 도구들을 사용함으로써 시스템 설계의 무결함이 증명되면 시스템이 구현되었을 때 결함이 발생하는 가능성을 낮출 수 있다. 그러나 이런 방법을 사용함으로써 하드웨어와 소프트웨어의 품질이 높아진다 하더라도 완전히 결함에서 자유로운 시스템을 구축하는 것은 쉽지 않으며, 특히 설계자가 의도적으로 삽입하여 도구로써 찾을 수 없는 보안 결함은 시스템의 의존성에 매우 큰 영향을 줄 수 있으므로 결함허용/침입감내 기법은 반드시 필요하다.

IV. 연구동향

1. 미국의 연구동향

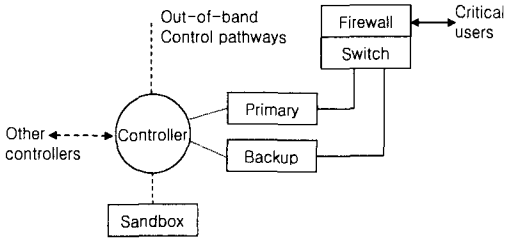
미국에서는 DARPA에서 다수의 침입감내시스템 관련 프로젝트들이 수행 중이다. 이 프로젝트 중 대표적인 것은 HACQIT[4], SITAR[5], Willow[6], ITUA[7] 등의 연구가 있다. 이 장에서는 HACQIT과 SITAR의 연구내용을 분석한다.

1.1 HACQIT

HACQIT(Hierarchical Adaptive Control of Quality of service for Intrusion Tolerance) 프로젝트는 UC Davis에서 진행하고 있는 것으로 사용자 성능 25% 이상의 저하를 방지하면서 네 시

간 동안의 침입 감내 제공을 목표로 하고 있다. 감내하고자 하는 대상으로는 소프트웨어 오류에 관한 것만을 다루고 있다. HACQIT 시스템 아키텍처는 그림 8과 같다.

HACQIT 시스템 내에는 두 대의 서버, 즉, primary와 backup이 존재한다. 이들 서버가 제공하는 서비스는 침입차단기능을 수행하는 게이트웨이를 통해서 사용자와 연결된다. 이들을 모니터하고 제어하는 controllers는 결함 진단 소프트웨어를 포함하고 있다. 그림에서 점선으로 나타낸 연결은 별도의 out-of-band 네트워크로 구성되어 일반 사용자가 접근할 수 없도록 되어 있다. Sandbox는 primary 및 backup 서버의 데이터 복제를 가지고 있어서, 두 서버의 결과가 일치하지 않는 경우 일시적인 문제인지, 공격에 의한 것인지를 파악하는데 이용된다. HACQIT는 오류 검출과 failure 차단을 위해 중복성과 다양성을 복합적으로 이용한다. 즉 한 개의 요청은 primary 뿐 아니라 backup 시스템에도 전달되며 Controller는 각 시스템에서 오는 응답을 비교하여 두 응답이 서로 같으면 고장이나 침입이 없는 것으로 간주한다. 만일 결과값이 다르면 고장으로 간주되어 더 이상의 피해가 확산되지 않도록 하기 위해 어떤 웹 서버가 공격받은 것인가를 판단하고 수리 및 복구 절차를 행한다.



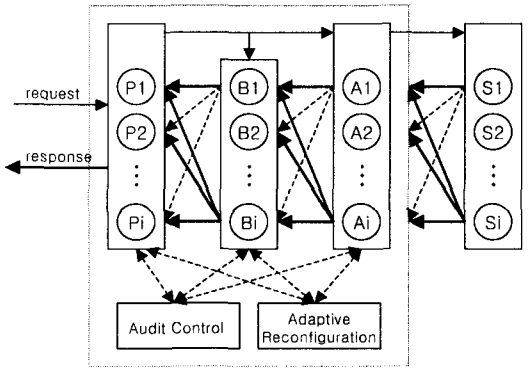
〔그림 8〕 HACQIT의 구조

이 시스템의 구조는 매우 간단하기 때문에 일반 COTS 서버들로 비교적 쉽게 구현할 수 있다. 다만 침입을 탐지하는 기능이 미약하고 확장하는데 한계가 있고, 또한 사용자 요청이 응용 서버에 바로 전달되지 않기 때문에 시간적 추가 비용이 존재한다. 그리고, Controller와 Sandbox가 외부에 노출될 경우 새로운 취약점이 될 수 있다.

1.2 SITAR

SITAR(Scalable Intrusion-Tolerance Architecture)는 MCNC라는 회사(www.mcnc.org)와 Duke 대학에서 공동으로 연구를 진행하고 있는데, 분산 서비스, 특히 COTS 서버를 위한 침입감내 구조를 제시하고 있다. 이 연구는 분산 서비스를 위한 침입감내시스템 구축을 위한 프레임워크를 제공하기 위해 취약성을 보강하고 필수 응용에 대해서는 언제나 최소한의 서비스 제공이 가능하도록 하는 것을 목적으로 한다.

이 시스템은 COTS 서버들이 침입에 취약한 것으로 가정한다. 이 시스템의 프락시 서버(P1..Pn)는 시스템이 제공하려는 서비스를 위한 public access point 구실을 한다. 새로 도착한 요청은 해당 COTS 서버들에게 전달되고 이 때 해당 Ballot Monitor (B1..Bn)들과 Acceptance Monitor (A1..An)들도 이 사실을 알게 된다. COTS 서버(S1..Sn)의 응답은 우선 Acceptance Monitor들에 의해 유효성 검사가 행해지고, 이어서 ARM과 같은 다른 모듈로부터 침입 발생 정보를 얻는다.



〔그림 9〕 SITAR의 구조

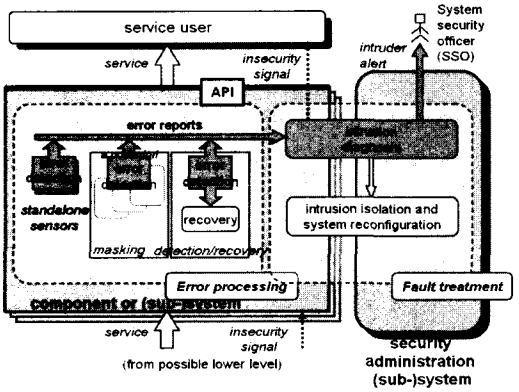
이 방안의 장점은 기존 COTS 서버를 바꾸지 않고 적용 가능하고 사용자에게도 투명하다는 것이다. 그러나 동시에 대량의 COTS 서버에 대한 공격 대응 보장의 한계와 COTS 서버 이외의 구성 요소들이 공격의 취약성을 갖고 있지 않아야 한다는 부담이 있다. 또한 침입 대응 과정을 담당하는 기능이 분산되어 있어서 과도한 지연 가능성이 있고 구성이 비교적 복잡하여 구현 비용이 높을 가능성이 있다.

2. 유럽의 연구동향

유럽에서는 MAFTIA[8](Malicious and Accidental-Fault Tolerance and Information Assurance)

라고 하는 침입감내기술 연구를 진행하였다. MAFTIA는 영국의 University of Newcastle upon Tyne을 중심으로 6개 국가에서 공동으로 연구하고 있는 프로젝트이다. 이 연구는 제목과 같이 악성 프로그램과 결함에 의한 결함허용과 정보보증을 목적으로 한다.

이 프로그램은 모두 6개의 workpage들로 구성되어 있다. 이들은 각각 시스템의 한 부분씩을 담당하며, 1) 시스템의 개념적 모델, 2) Middleware, 3) 침입탐지, 4) 신뢰성있는 3자 시스템, 5) 분산 인증, 6) 검증과 평가로 나뉜다. 그림 10은 MAFTIA의 구조를 나타낸 것이다.



〔그림 10〕 MAFTIA의 구조

V. 맺음말

침해사고를 예방하고 효과적인 대응방법을 마련하기 위하여 침입차단기술, 침입탐지기술 등 여러 가지 정보보호기술들이 개발되고 있다. 그러나 이와 같은 기술들은 알려진 취약점에 대한 예방과 탐지에 대하여는 좋은 결과를 보여주지만, 알려지지 않은 취약점이나 공격에 대하여는 적절한 대응이 쉽지 않은 단점이 존재한다. 그러므로 이와 같이 알려지지 않은 취약점이나 공격에 의한 침해사고를 방지하기 위한 기술이 필요하며, 침입감내시스템이 이에 대한 한 가지 해결책으로 제시될 수 있다.

침입감내시스템은 시스템에 대한 악의적 공격이 발생하여도 일정한 수준이상의 서비스를 지속적으로 제공할 수 있도록 고안된 시스템이며, 이러한 시스템은 시스템의 의존성 특징을 만족시키기 위한 여러 가지 기술을 적용함으로써 구축될 수 있다. 시스템의 의존성을 만족시킬 수 있는 기술들은 이미 결함허용기술을 통하여 연구되어 왔으며 이러한 결함허용기술을 효과적으로 사용하고 악의적 공격에 대응하기 위한 기술을 추가함으로써 침입감내기술이 개발될 수 있을 것으로 보인다.

이러한 침입감내기술은 급속하게 늘어나고 있는 취약점과 이로 인한 보안사고들에서 정보통신기반이 생존할 수 있도록 함으로써 시스템의 신뢰성과 안정성을 매우 높여줄 수 있을 것으로 기대된다.

참 고 자 료

〔1〕 Fray, J.-M., Deswarte, Y. and Powell, D., Intrusion-Tolerance using Fine-Grain Fragmentation-Scattering, in *Symp. on Security and Privacy*, Oakland, CA, USA, pp.194-201, 1986

〔2〕 J.C. Laprie, *Dependablity:Basic Concepts and Terminology*, Springer Verlag, 1992

〔3〕 B. Randell, Dependability - A Unifying Concept, *Computer Security, Dependability, and Assurance: From Needs to Solutions*, pp.16-25, 1008, IEEE CS

〔4〕 J. Just, *et.al.*, Intrusion Tolerance through Forensics-Based Attack Learning, Proc, of the ICDSN 2002 Supplementary Vol., p.C-4-1, June 2002, IEEE CS

〔5〕 F. Wang and C. Killian, Design and Implementation of SITAR Architecture: A Status Report, *Proc, of the ICDSN 2002 Supplementary Vol.*, p.C-3-1, June 2002, IEEE CS

〔6〕 J. Knight, *et.al*, The Willow Architecture: Comprehensive Survivability for Large-Scale Distributed Applications, *Proc, of the ICDSN 2002 Supplementary Vol.*, p.C-7-1, June 2002, IEEE CS

〔7〕 T. Courtney, *et.al*, Providing Intrusion Tolerance with ITUA, *Proc. of the ICDSN 2002 Supplementary Vol.*, p.C-5-1, June 2002, IEEE CS

〔8〕 Paulo Verissimo, *et. al.*, The Timely Computing Base: Timely Actions in the Presence of Uncertain Timeliness, *Proc. of the ICDSN 2000*, pp.533-542, IEEE CS

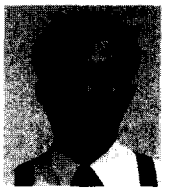
〈著 者 紹 介〉

최 중 섭(Choi, Joongsup)

1993년 : 인천대학교 공과대학 전자계산학과 공학사
1995년 : 숭실대학교 대학원 컴퓨터학과 공학석사
2000년 : 숭실대학교 대학원 컴퓨터학과 공학박사
1995년~1996년 : 한국전산원 초고속사업단 연구원
2000년~현재 : 한국정보보호진흥원 기술단 선임연구원
관심분야 : 컴퓨터보안, 내장실시간시스템, 분산시스템

이 경 구 (Lee, Koung-Goo)

1982년 : 한양대학교 공과대학 무기재료공학과 공학사
1986년 : University of Central Arkansas Computer Sceience Department
1988년 : University of Arkansas Computer Science Department MS
1996년 : Kent State University Computer Science Department PhD
1996년~현재 : 한국정보보호진흥원 기술단 시스템기술팀장
관심분야 : 정보보호, 보안성평가, 정보보증



김 홍 근 (Kim, Hong-Geun)

1985년 : 서울대학교 컴퓨터공학과 공학사
1987년 : 서울대학교 컴퓨터공학과 공학석사
1994년 : 서울대학교 컴퓨터공학과 공학박사
1994년~1996년 : 한국전산원 선임연구원 전산망안전보안센터장
1996년~현재 : 한국정보보호진흥원 책임연구원 기술단장
관심분야 : 컴퓨터 보안, 병렬 알고리즘