Journal of Korea Multimedia Society Vol. 7, No. 1, January 2004(pp. 113-125)

# 능동보안 아키텍쳐를 위한 컴포넌트 분류 및 명세방법

김상영[†], 김재웅[††], 황선명[†††]


## 요 약

능동 네트워크는 능동 네트워크 애플리케이션의 통신 중 요구사항의 처리를 할 수 있게 하는 소프트웨어 프레임워크를 제공한다. 능동보안을 위한 컴포넌트 아키텍쳐는 관련 컴포넌트들의 조합으로 재사용 시스템을 쉽게 구축할 수 있다. 이 아키텍처는 컴포넌트를 획득하고, 이해하며 조립하기 위한 표준 계층으로서 컴포넌트 식별, 탐색과 조정을 위한 지침을 제공해야 한다. 본 논문에서는 최근 연구되어지고 있는 능동네트워크의 보안에 대한 부분을 관리하기 위한 능동보안 컴포넌트 개발을 위한 아키텍쳐 설계 및 도메인 분석을 하였으며, 능동보안 도메인 아키텍쳐를 이용한 컴포넌트 개발을 위한 설계명세에 대하여 연구하였다.


# Component Classification and Specification on Active Security Architecture

Sangyoung Kim[†], Jaewoong Kim[††], Sunmyung Hwang[†††]


## ABSTRACT

Active networks aim to provide a software framework that enables active network applications to customize the processing their communications. Active security component architecture focuses on the support of reuse system by active security component. The architecture is standard layer to acquire, understand, and assemble component, and it has to support a guideline for component identification, search and customization. In this paper we present the active security architecture as a standard model of discrete Active network solution, and we propose the method for component classification and specification.


**Key words:** Component(컴포넌트), Active Network(능동 네트워크), Component Architecture(컴포넌트 아키텍처), Component Domain(컴포넌트 도메인)

## 1. 서 론

21세기에 들어 전자상거래, B2B, e-Business 등 다양한 형태의 전자적 업무처리가 실용화되면서 사회의 제반활동 구조에 큰 변화를 가져왔다. 이러한 사회활동과 더불어 보안 위협요소 역시 더욱 증가하였고, 이러한 위협요소를 방어하기 위한 보안 기술에 대한 연구개발이 대단히 중요해졌다. 그러나, 각 응용분야별 보안기술의 개발은 이기종간 호환성의 결여로 인한 중복된 투자 손실 등의 문제점이 발생되었다. 이러한 문제점을 해결하기 위하여 범용적 사용이 가능한 표준의 개발이 지속적으로 요구되었고, 이를 위하여 보안 API(Application Program Interface)가 설계되었다[1,11].

이에 따라 소프트에어의 품질을 보증하고 재사용을 통한 소프트웨어 개발 생산성을 향상시키기 위한 방법이 요구되었고, 이를 수용하기 위한 방법으로 소

※ 교신저자(Corresponding Author) : 김상영, 주소 : 대전광역시 동구 용운동 96-3(300-716), 전화 : 042)280-2544, FAX : 042)284-0109, E-mail : jayusop@zeus.dju.ac.kr
접수일 : 2003년 2월 12일, 완료일 : 2003년 6월 5일
[†] 대전대학교 대학원 컴퓨터공학과 박사과정
[††] 정회원, 공주대학교 멀티미디어정보 · 영상공학부 부교수 (E-mail : jykim@kongju.ac.kr)
[†††] 정회원, 대전대학교 컴퓨터공학과 교수 (E-mail : sunhwang@dju.ac.kr)
본 연구는 한국과학재단 목적기초연구(R01-2001-000-00343-0 (2003)) 지원으로 수행되었음.

프트웨어를 부품화하고 이를 조립, 합성하여 애플리케이션을 개발하는 컴포넌트 기반의 개발 방법이 등장하였다. 컴포넌트 기술의 등장으로 이미 생성된 컴포넌트를 조립하여 제작함으로써 소프트웨어 개발의 적시성 및 생산성을 향상시킬 수 있으며, 다양한 제품간 규격화 및 표준화 등을 유도할 수 있게 되었다.

이러한 컴포넌트 설계기법을 범용 암호서비스 개발에 적용한다면 암호서비스에 대한 상호운용성의 확보와 자체검증의 용이성, 이기종 시스템간 상호 호환성, 필요한 모듈의 추가 및 변경 용이성 등의 장점을 갖는다. 표준화된 암호 컴포넌트 인터페이스는 각종 애플리케이션에서 공통적으로 사용되는 모듈성 및 내구성을 갖기 때문에 동일한 보안 서비스의 중복된 개발노력을 감소시키고, 개발비용의 절감 및 신뢰성 향상의 이점이 있다.

## 2. 능동보안

### 2.1 능동 보안의 정의 및 범위

네트워크 공격 기법의 다양화 및 지능화함에 따라 국가적으로 중요한 정보통신망에 대한 사이버 공격 위협이 증대하고 있다. 그 특징으로는 분산 환경에서 다수 공격 에이전트를 이용하여 특정 상용 서버의 서비스를 제공을 마비시키는 분산 서비스 거부 공격의 출현과 해외 해커들의 국내 전산망을 우회 루트로 활용한 사례의 증가 등 사이버 공격 행위가 점차 범죄의 강력한 주요 수단으로 이용되는 추세에 있다.

현재의 네트워크 보안 관리는 방화벽, VPN, IDS 등 개별 기능이나 개별 제품 중심으로 지역적인 네트워크에 적용되고 있으며, 네트워크 관리도 운용자의 수동적인 방법에 의존하므로 네트워크가 대규모화 됨에 따라 제어/관리 방법의 복잡도가 증가하고 있다. 또한, 다양한 유형의 개별적인 보안 시스템이 적용되어 시스템간의 연동이 불가능하고, 새로운 보안 기능 추가 시에 하드웨어 및 시스템의 교체가 수반되는 등 이종의 보안장치간, 이종의 네트워크간, 이종의 사업자간의 상호 연동형 보안 서비스 환경을 제공할 수 없으며, 사이버 공격에 대응한 사용자 종단간의 안전하고 효율적인 보안 관리가 불가능하다.

따라서, 사용자 요구에 대응한 서비스 품질의 다양화 및 특화된 고객 지향 서비스를 지원하고 또한 고객의 정보, 서비스의 품질을 보호하기 위한 보안 기술의 개발이 필요하다. 이를 위해 네트워크 관리 기능에 프로그래밍 가능성(programmability)과 서비스 온 디맨드(service on demand)라는 특성을 갖는 능동 네트워크의 기술을 시스템 및 네트워크 보안관리 기능과 접목시킴으로써, 기존의 소극적인 네트워크 관리, 네트워크 해킹 기술에 비해 더딘 대응 및 정적인(Static) 통합관리 솔루션의 문제점을 능동적으로 해결하고, 기능 업그레이드를 신속히 할 수 있는 능동 보안관리 기술 개발이 요구된다고 할 수 있다.

능동 보안 기술은 이런 보안 환경의 변화 아래에서 현재의 네트워크 보안이 가지는 문제점을 해결하기 위해 미국을 중심으로 연구가 진행 중인 차세대 네트워크 보안 기술이다.

능동 보안 기술이 가져야 할 특징은 다음과 같다[1,3].

- 능동형 보안 기술은 하부 망의 구조 및 종류, 시스템의 종류 및 동작 환경에 무관할 정도로 유연할 실행 구조 및 보안 체재를 유지할 수 있어야 한다.
- 특정 조직의 관리 도메인 상에 적용되는 기술이 아니라 여러 조직의 관리 도메인들에 걸쳐서 해당 보안 기능을 수행할 수 있는 메커니즘을 제공하여야 함. 즉 Inter-domain까지 보안 관리 영역을 확대할 수 있어야 한다.
- 실시간적 모니터링 및 침입 탐지/대응을 수행할 수 있어야 한다.
- 네트워크를 통하여 보안 메커니즘이 자동으로 생성·복제·소멸되고, 능동적으로 보안 응용 서비스의 프로그래밍이 가능하여야 한다.

#### 2.1.1 능동보안기술 정의

능동 보안 기술이란 네트워크를 이용하는 사용자의 안전하고 효율적인 보안 관리를 위하여 프로그래밍이 가능한 보안 관리 메커니즘을 생성하고, 이를 이동형 센서 기술을 통해 네트워크의 노드에 전달하여 신속하고 능동적인 보안 관리 서비스를 제공하는 제반기술로써, 능동 보안 프레임워크, 이동형 센서 엔진 및 능동형 보안 관리 기술을 포함하는 것을 말한다[1,2,6].

이동형 센서 엔진은 네트워크 노드, 호스트, 보안장치에 탑재될 수 있으며, 하드웨어나 운영체제에 독립적이고, 안전한(safe) 실행 환경을 가지고 있어서, 일종의 액티브 패킷인 센서에 담겨져 있는 능동 보안

메커니즘을 실행할 수 있는 보안 실행 엔진이다.

2.1.2 핵심 요구 사항

능동 보안 기술을 구현하기 위해서는 여러 가지 다양한 요구 사항이 있으나 핵심 요구 사항은 다음과 같다[6].

- 서비스의 확장성

능동 보안 서비스는 다양한 유형의 사용자와 보안 서비스를 수용하기 위해서 유연한 확장성을 지원해야 한다.

- 신속성

사이버 공격에 따른 침입에 대한 실시간적인 대응을 가져야 하며, 대응 결정을 위해 자동적인 또는 수동적인 제반 정보의 분석이 선행되어야 한다.

- 이동성(mobility)

보안 기능, 보안 응용, 보안 서비스, 보안 프로토콜, 수행환경(Execution Environment) 등을 목적하는 네트워크의 노드로 항행(migration)할 수 있어야 하며, 이동하고자 하는 정보는 가능한 경량화(light-weight)해야 한다.

- 적응성(adaptability)

보안 기능 또는 보안 서비스가 제공될 때 기존에 설치되어 활용중인 응용 또는 서비스를 통해서 적절히 요구할 수 있어야 하며, 가능하면 변경 없이 기존 기능들을 이용함으로써 기존의 투자를 최대한 보호할 수 있어야 한다.

- 네트워크 노드 자원의 재구성 가능성(re-configurable)

네트워크 보안관리를 위해 설정된 보안 정책에 따라 네트워크 자원을 동적으로 재구성하고, 다수의 네트워크 구성 요소를 일시에 제어할 수 있도록 해야 한다.

- 보안 관리 영역간의 연계(correlation)

사이버 공격에 따른 침입에 대한 단독 ISP(Internet Service Provider)내 뿐 아니라, ISP간의 연계에 의한 글로벌 네트웍 차원에서의 대응 방안을 제공해야 한다. 이때, ISP내의 망관리 시스템, 정책 서버, 인증 서버, 네트워크 기반 보안 서버 등과의 정보 교류를 고려할 수 있다.

- 동일 플랫폼 상의 다양한 응용 및 서비스

- 다양한 형태의 보안 응용 또는 서비스를 센서 엔진이라는 보안 실행 플랫폼상에서 제공할 수 있는 구조의 유연성을 갖추어야 한다.

- 능동형 공격 대응(active response)

사이버 공격에 대해 방어 수준의 소극적인(passive)대응에서 벗어나 공격자의 호스트를 색출하여 역으로 보복 공격까지도 수행할 수 있는 능동적인 대응 방안을 고려해야 한다.

- 목적 지향 보안 서비스(mission-oriented security service)제공

사용자가 요구하는 특정 목적에 적합하도록 맞춤형 보안 서비스 네트워크(customized service network)의 구성이 가능해야 한다.

## 2.2 능동보안의 기술적 구조

능동 보안 기술은 새로운 공격 기법 및 탐지 대응 기술의 등장과 같은 보안 환경 변화에 유연하게 적응할 수 있어야 하고, 기존 네트워크 보안 기술에 비해 능동적이고 공격적인 대응이 가능하기 위해서는 전체적인 보안 체계에서의 구조와 보안 체계를 실행하기 위한 실행 환경 상의 구조로 나누어 고려할 수 있다.

보안 체제를 구성하는 요소로는 네트워크 노드나 호스트를 이동하여 해당 노드에 연결설정하여 관련 능동 보안 기능을 수행하는 이동형 센서와 이동형 센서에게 하부 시스템에 독립적인 실행 환경을 제공하기 위한 이동형 센서 엔진(보안 네트워크 엔진), 전체적인 능동 보안 도메인을 관리하고 해당 도메인 상에서 능동 보안 관련 기능을 조율한 능동 보안 관리 시스템으로 구성된다.

능동 보안 기능을 구현하는 기능 요소들은 침입 탐지, 공격자 추적, 대응, 센서들의 관리 및 모니터링을 담당할 각종 센서들로 구성된다. 센서들은 자신이 담당하고 있는 기능의 특성 및 기능이 수행될 필요성이 있는 위치에 대한 정보를 바탕으로 해당 보안 도메인을 구성하는 노드들 사이를 이동하여 해당 기능을 수행하여, 필요에 따라 복제, 생성, 소멸하게 된다. 보안 네트워크 엔진은 이런 이동형 센서들에게 하부 시스템의 구조에 독립적인 수행환경을 제공하게 된다.

센서들은 이동형 코드로써 구성되며 센서를 전체 도메인 상에서 운용하기 위해 필요한 지원 기능으로는 센서의 위치 및 주소 관리 기능, 센서간의 통신 프로토콜의 개발, 센서 기능 정의를 위한 프로그래밍 언어, 도메인 상에서의 센서의 재구성 및 장애 복구

기능이 필요하다. 또한 센서는 보안 네트워크 엔진을 통하여 해당 노드의 자원이나 제어권을 이용하게되므로 이동형 센서 자체에 대한 보안 기능이 필요하다.

능동 보안 관리 시스템은 해당 도메인 상에서의 능동 보안과 관련된 관리 기능을 수행하여, 센서들에 의해 각 노드에서 행해진 결과들을 전체적으로 조율하여 최적의 결과를 도출하는 기능을 수행하며 이에 필요한 요소기술은 그림 1과 같다[1,12].

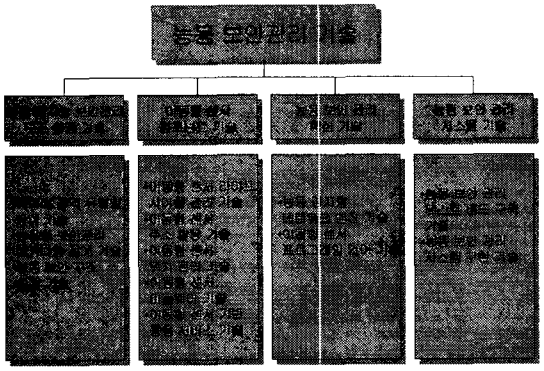


그림 1. 능동 보안 기술의 요소 기술

## 3. 능동 보안 컴포넌트 아키텍쳐

### 3.1 능동보안 컴포넌트 아키텍쳐 개요

컴포넌트 사용에 있어 개발자들은 소프트웨어 재사용의 초기 과정에서 겪었던 것과 같이 도메인 컴포넌트의 식별과 검색, 적용에서 비용적, 품질적 위험부담을 고려해야만 한다. 특히 표준 아키텍쳐를 따르는 명세화 및 조직화의 미흡은 능동보안 컴포넌트 개발 시 프로세스의 진행에서 저조한 생산력을 야기시키는 근본 원인이 된다.

현재의 컴포넌트 아키텍쳐들은 객체 기반의 시스템 플랫폼 중심의 벤더 의존적인 한정적인 구조로서, 일반화된 규칙을 제공하지 못하고 있으며 특별한 분야인 능동보안에 대하여는 구체적인 체계가 정해져 있지 않다. 컴포넌트의 의미 또한 특정 영역에서 실행성의 객체지향 응용 시스템 혹은 이를 이루는 개별 객체를 나타내는 것으로 컴포넌트 아키텍쳐로서 적용하기에는 많은 문제가 남아있다.

따라서, 컴포넌트 사용자들이 능동보안을 위한 최상의 비즈니스 솔루션을 갖는 컴포넌트를 선택할 수 있도록 컴포넌트의 명확한 의미를 포함하는 컴포넌트 이용 명세와, 컴포넌트 개발자들이 비즈니스 요구를 수용하여 유통체제에 적극 대응할 수 있는 컴포넌트 개발 명세들이 아키텍쳐 참조 모델에 기반하여 제공되어져야만 한다.

컴포넌트 아키텍쳐는 능동보안에 관련된 다른 종류의 컴포넌트들을 연관시키기 위한 표준 계층으로 컴포넌트의 획득, 이해, 조립을 위한 레이아웃을 제시함으로써 사용자들이 필요로 하는 컴포넌트들을 식별하고, 검색하며 커스터마이즈할 수 있는 가이드라인을 제공해야한다. 따라서 현재의 시스템 개발 환경과 연관하여 다음과 같은 요구 사항을 가진다[10].

1) 능동보안 솔루션 개발자에 대한 투명한 가이드라인 제공
2) 분산 컴퓨팅을 위한 멀티 벤더의 멀티 솔루션 통합을 위한 방법 제공
3) 기본 인터페이스를 공유하고 하위 계층 컴포넌트 조합에 의한 상위 계층의 독립적 응용 및 그룹화된 능동보안 컴포넌트 형성
4) Scope와 Abstraction에 따른 수평적 컴포넌트 계층과 동일 영역의 입자성(Granularity)에 따른 수직적 컴포넌트의 범주 형성

본 논문에서는 분산 컴퓨팅 환경 하에서 능동보안 비즈니스 솔루션을 위한 표준 모델로서 능동보안 컴포넌트 아키텍쳐를 제시한다. 본 아키텍쳐는 Sanfrancisco를 기반으로 멀티 벤더/멀티 솔루션의 통합을 위해 컴포넌트의 스코프와 추상성, 입자성을 기준으로 계층적 분류를 시도한다.

다음은 능동보안 아키텍쳐 정의를 위한 원칙이다.

1) 능동보안 비즈니스영역을 위한 컴포넌트와 일반적인 시스템 요구 컴포넌트로 구분하고 기반이 되는 필수 컴포넌트와 부가적인 선택적인 컴포넌트로 구분한다.
2) 구분되어진 내용을 개별 능동보안, 능동보안 관리, 관리 응용의 3개의 레이어로 정의.
3) 완성되어 수정이 불가한 컴포넌트와 패턴 형식의 커스터마이즈가 가능한 컴포넌트로 분류하며 컴포넌트 제공 서비스의 범주에 따라 계층적 관계를 형성하는 군을 정의.
4) 기존에 정의된 분산 서비스의 컴포넌트를 계층에 포함.

### 3.2 능동보안 컴포넌트 아키텍쳐 적용 목적

능동보안 소프트웨어 부품화를 통한 재사용의 실현을 위한 기존 연구들은 소프트웨어 부품들의 추상성과 입자성 그리고 독립성과 구현성을 주요 카테고리 하여 수행해 왔다. 즉, 방법론적인 측면에서 절차적 프로그래밍의 기능 모듈에서부터, 객체지향 방법론에서 소스 코드 단위의 독립 패키지인 클래스에 이어, 현재 컴포넌트 기반 방법론(CBD : Component Based Development)에서의 컴포넌트에 이르기까지 그 대상물들의 적절한 생성과 획득, 효과적인 적용이 주요 접근 방법으로 제안되어 왔다. 특히 컴포넌트는 명확한 인터페이스에 바탕을 둔 비즈니스 로직의 모듈성과 이에 대한 커스터마이즈 및 실행성을 특징으로 완벽한 블랙 박스와 실행시의 조립을 통해 이상적 솔루션을 창출할 수 있는 것으로 기대되고 있다[8,10].

표 1. 컴포넌트 아키텍쳐의 특징

| |
|---|
| ① 관련된 다른 종류의 컴포넌트들을 연관시키기 위한 표준 계층 |
| ② 기본 인터페이스를 공유하고 하위 계층 컴포넌트 조합에 의한 상위 계층의 독립적 응용 및 그룹 컴포넌트 형성으로 응용 구축을 위한 (비즈니스, 재사용) 프로세스 제공 |
| ③ Scope와 Abstraction에 따른 수평적 컴포넌트 계층과 동일 영역의 입자성에 따른 수직적 컴포넌트의 범주 형성 |
| ④ 컴포넌트의 생산, 배포, 획득, 조립 및 멀티솔루션 통합을 위한 레이아웃 제시 |

Prieto-diaz는 도메인 분석을 새로운 시스템을 만들어낼 때 재사용 가능한 것을 만들 목적으로 식별되고, 인지하며, 체계화되어진 소프트웨어 시스템을 개발하는데 사용되는 정보에 바탕을 둔 프로세스 라고 처음으로 정의하였다[13]. FODA와 같은 초기의 도메인분석 방법은 코드 지향적이었기 때문에 이해하기 힘들었지만, 근래에 ODM(Organizational Domain Modeling)과 같은 방법은 소프트웨어 개발의 모든 단계에 걸쳐 고려된다.

컴포넌트를 분류하기 이전에 도메인에 대한 이해와 식별을 위해 도메인 프로세스를 고려해야 한다. 도메인 프로세스는 도메인 지향 프로세스와 애플리케이션 지향 프로세스로 나누어진다. 도메인 지향 프로세스에서 얻은 정보는 도메인 모델로 정의되며, 이것은 애플리케이션 지향 프로세스에 주요한 자원이 된다. 프로세스의 초기자원은 기존의 시스템과 요구사항을 기반으로 도메인 전문가의 지식이 되며, 기초적인 이론과 최근의 기술, 도메인에 관련된 정보를 식별하고 수집, 체계화하고 나타내는 과정으로 전개해 나간다.

- 도메인 지향 프로세스

기술적인 문서, 기존의 시스템 고객의 관점, 경험, 현재와 향후에 요구될 사항을 포함한 자원으로부터 도메인에 대한 정보를 수집한다. 이는 도메인 추상화와 폭넓은 기능적인 요구사항을 식별하기 위한 지식이 된다. 다음 단계로 첫 번째 단계에서 수집된 정보에 의해서 추상화단계에 도메인을 분류하고 구조화하는 것이다.

이 결과는 도메인 분류와 분류법에 유용한 정보로 사용되며 애플리케이션 도메인에서 범위를 식별하는데 도움을 준다. 도메인 컴포넌트를 식별하는 단계는 범위에 의해 분류되어지고 도메인 분류에 포함되는 핵심으로 그룹화 된다. 이 컴포넌트의 분류단계에서 또한 일반화-특수화 관계에 의해서 컴포넌트간에 관련성을 보이기 위해 일반화된다. 각각의 컴포넌트의 의미는 정의되어지고 컴포넌트의 명세에 첨가된다. 다음으로 도메인 지식 소스로부터 수집된 도메인 자원을 명세하는 것으로써, 도메인 자원은 도메인 사전, 설계 근거, 범위에 의해 도메인 모델에 포함된 시나리오를 포함한다. 범위에 대해서 조정하는 것으로, 자원은 그 범위에 명세되어진 컴포넌트에 대한 비기능적인 요구사항으로써 사용된다.

결과적으로 컴포넌트간의 관련성은 참조 아키텍처로써 도메인 모델로 첨가되어지고 다음 절에서 모델 되어진다. 또한, 참조 아키텍처는 컴포넌트의 추적성을 쉽게 하는 컴포넌트간의 관련성과 의존성을 나타낸다. 프로세스에서 나머지 부분은 컴포넌트간의 연결과 잘못된 영역에서 명세 되어진 컴포넌트를 식별하기 위해 도메인 아키텍처를 검사한다. 프로세스는 안정적인 도메인 모델이 될 때까지 반복한다.

- 애플리케이션 지향 프로세스

새로운 애플리케이션은 도메인 분류법을 사용하여 통합되어진 도메인 범위를 식별하는 것으로 시작한다. 범위가 정해지고 재사용 가능한 컴포넌트의 수는 주요 도메인에서 식별되고 그룹화된 컴포넌트

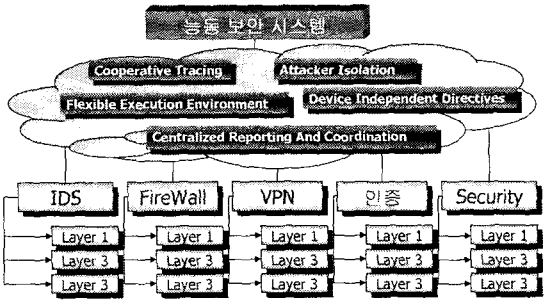
에서 참조될 수 있다. 사용자의 요구사항은 도메인 시나리오를 사용하여 식별되어 질 수 있다. 이 과정을 반복하는 동안 일반적인 아키텍처를 사용하여 추적 가능한 다른 컴포넌트의 의미적인 명세에 도움을 준다. 또한, 시스템의 기능적인 요구사항 명세에서 볼 수 있는 서비스를 구성하는데 사용된다. 모든 컴포넌트가 반복할 때 시스템은 도메인의 비기능적인 요구사항에 의해서 통합된다.

프로세스의 마지막 단계는 전체 재사용 결과가 평가된 것에 대해 재사용성을 지평가한다. 이것은 두 가지 형태의 작업이다. 첫 번째는 어떤 성공적인 재사용과 존재하는 도메인 결과물에 밀접한 문제를 평가한다. 두 번째는 현재의 성과로부터 나타날 수 있는 재사용에 대한 새로운 가능성을 조사한다. 이것은 새로운 컴포넌트, 시나리오, 이론적 근거, 사전이나 존재하는 결과에 적응이다. 평가된 결과는 수정된 도메인 모델과 모든 가능한 재사용성이 조사될 때까지의 작업을 의미한다.

능동보안 시스템에서 요구되어지는 특징들은 협력적 추적(Cooperative Tracing), 공격자 차단(Attacker Isolation), 유연한 실행 환경(Flexible Execution Environment), 장치에 독립적인 명령(Device Independent Directives), 집중화된 보고 및 협조(Centralizes Reporting & Coordination) 의 5가지를 추출할 수 있다. 이러한 요구되어지는 특징들은 IDS, FireWall, VPN, 인증, Security의 5가지의 서브 도메인으로 분류할 수 있다. 각각의 서브도메인은 컴포넌트의 규모에 기능에 따라 총 3개의 소규모 레이어로 분류하였으며 각각의 레이어들의 특징은

- Layer 1 : 단위기능 비즈니스 로직으로서 기능은 수행 하지만, 각각의 컴포넌트는 능동보안의 기능을 수행하지 않는 컴포넌트를 분류.
- Layer 2 : Layer 1의 컴포넌트 보다 크고 능동보안 관련 기능을 수행 가능한 컴포넌트의 분류.
- Layer 3 : Layer 2 의 컴포넌트들의 조합으로서 능동보안 비즈니스를 수행가능할 수 있는 컴포넌트의 분류.

이러한 능동보안의 특징들을 이용하여 능동 보안 도메인을 분석 및 분류하면 그림 2와 같이 능동 보안 도메인을 분류할 수 있다.




- Layer 1: 개별적인 단위 보안 컴포넌트
- Layer 2: 보안 관리 컴포넌트
- Layer 3: 관리 응용 컴포넌트

그림 2. 능동 보안 도메인 분류

### 3.3 컴포넌트 분류 코드 정의

응용 개발자들이 자신의 시스템에 적절한 재사용 가능한 자산 즉, 컴포넌트를 식별, 저장하고 필요할 때 검색하며 기능적으로 관련된 컴포넌트 사이를 브라우징하고 네비게이터 할 수 있기 위해서는 컴포넌트 분류 체계의 제공은 필수적이다. 따라서, 컴포넌트 자산들을 저장시킨 후 요구가 발생할 때마다 검색, 이해를 통해 재사용을 실현하는 저장소 시스템은 컴포넌트의 인덱싱의 기본이 되는 분류 코드 체계로부터 시작할 수 있다. 그래서 능동보안 컴포넌트 명세를 위한 분류코드 코드화 규칙에 대하여 연구한다.

#### 3.3.1 능동보안 컴포넌트 분류 코드화 규칙

현대 분류 체계는 열거식 체계와 조합식 체계로 구분되고 있다. 열거식 체계를 사용하는 주된 이유는 유용성이나 편리성 때문이지만 이 체계는 합성주제나 복합 주제의 표현이 어렵고 자료에 내포되어 있는 주제에 대한 심층적인 분석이 어렵다. 조합식 체계는 패싯 구조를 사용하여 모든 주제에 대한 분류가 가능하도록 하고 있다. 이 영향으로 기존의 열거식 체계에서도 조합식의 구조적 요소를 많이 수용하고 있는 상황이다.

효율적인 분류를 위해서는 잘 정의된 분류 체계가 제공되어야 한다. 잘 정의된 분류 체계란 분류 대상의 종류와 용도별 특성을 반영한 계층구조 항목들로 구성된다. 이러한 항목들을 분류 컴포넌트라 하며 이들은 분류 체계를 구성하는 단위 요소들이다. 많이 쓰이고 있는 분류 체계로는 듀이식 십진 분류법(DDC), 국회 도서관 분류법(LCC) 등이 있으나 이들

분류 체계는 대부분 문헌 분류의 목적으로 사용되고 있다. 이들 분류 체계를 소프트웨어 분류에 적용하는 데는 많은 문제가 있다. 특히 소프트웨어는 물리적 특성이 없는 대상이므로 문헌과는 차이가 있고 정보의 용도가 컴퓨터 상에서 실행된다는 측면에서 소프트웨어의 특성을 적절히 반영할 수 있는 분류체계를 필요로 하게 되었다. 소프트웨어 컴포넌트 분류 체계는 컴포넌트가 갖는 주제 혹은 개발된 용도에 따라 나열될 수 있는 모든 영역에 대해 고유한 분류 번호를 부여한 분류 번호의 집단이다. 따라서, 분류 번호는 컴포넌트의 영역 특성을 충실히 반영한 형태이므로 올바르게 부여된 분류 번호를 가진 컴포넌트라면 이용자는 분류 번호를 통해 해당 컴포넌트의 특성을 판별할 수 있다.

분류 표기법은 개발되고 수집된 컴포넌트에게 식별될 수 있는 고유의 코드를 부여하고 체계적으로 분류하기 위해서 고려되어야 할 부분이다. 도메인 프로세스를 통해 식별된 기능적인 요소와 비기능적인 요소 두 부분으로 코드체계를 나누어서 표기하도록 한다. 기능적인 요소는 구성된 컴포넌트 참조 모델을 기반으로 대분류, 중분류, 소분류로 구성하며, 비기능적인 요소는 패싯으로 구성된 11개의 어휘를 기반으로 중복되지 않고 일관된 형식을 최대한 유지하며, 융통성을 가지기 위해 필요에 따라 영문자와 코드의 길이를 고려하여 표기법을 작성하였다.

기능적인 요소와 비기능적인 요소는 구별하기 쉽게 "-"로 분리시켜서 표기하도록 하였으며, 기능적인 요소는 분류 리스트에 컴포넌트 명칭과 함께 표기하며, 비기능적인 요소는 개발되거나, 수집된 실제 컴포넌트를 등록할 경우에 다른 컴포넌트와 구별될 수 있는 유일한 코드를 포함하여 전체 표기법을 부여하도록 한다. 따라서, 기능적인 요소 표기법은 기본적으로 부여되며, 동일한 영역의 컴포넌트의 구별은 비기능적인 요소 표기법으로 가능하다.

본 논문에서 제시하는 능동보안 컴포넌트 아키텍쳐에 따른 명세 분류 코드는 다음과 같은 코드화 원칙을 적용하여 작성한다.

① 아키텍처 4계층을 대분류로 표기하며 각 계층은 영역별로 중분류 재표기
② 중분류는 규모에 따라 상세하게 세분화하며 이하 하이폰으로 소분류를 표기
③ 소분류 각각의 하이폰은 기능적인 하위 레벨의 단계를 표시
④ 중분류와 소분류는 각각 세 자리로 정하며 하이폰(-)으로 연결 표시. 단, 도메인 컴포넌트의 중분류는 다양하고 광범위함으로 특별히 지정
⑤ 소분류는 4단계 이하로 기능성 규모가 소입자(fine granularity)로 레벨링
⑥ 중분류 및 소분류의 잠재적 의미는 컴포넌트 기능별 규모 표시
⑦ 영역의 공통 부분 표현 위해 중분류와 필요 부분에 "00"은 공용으로 정의
⑧ 각 영역마다 그룹화하기 모호한 컴포넌트들은 "99"로 표기하여 기타로 정의

따라서 컴포넌트 표기법은 다음과 같은 식으로 표현된다.

대분류+중분류+보조도메인분류+{ .소분류}

구체적인 컴포넌트 코드화 방법은 10진 분류 코드에 기반하여 점증적인 레벨링과 자연 언어의 의미적 요소를 결합한 것으로 다음과 같은 방법으로 설명된다.

가. 대분류

능동보안 컴포넌트 아키텍쳐에 준하여 코드를 부여하며 공통 컴포넌트는 공통 비즈니스와 핵심 비즈니스로 확장하여 코드화한다. 도메인 영역의 대분류는 "D"로 시작하되 중분류를 위해 두 번째 자리는 "xx"로 표기한다.

Dxx : Domain Component
CBZ : Common Business Component
COB : Core Business Component

나. 중분류

대분류 이하 세자리로 각 계층 컴포넌트의 관련성에 따라 분류한다. 도메인 컴포넌트의 중분류는 대분류에 포함된 자리를 차지하며 공용과 기타는 자릿수를 고려하여 각각 "00"과 "99"로 표기한다. 도메인 컴포넌트 층은 알파벳으로 코드가 중복되지 않도록 설정하며 최초 식별된 중분류는 기타 코드 부여한 후 컴포넌트 범위가 확장될 경우 유사 컴포넌트와 합병하여 새로운 코드를 부여한다. 예를 들어 능동보안 도메인 컴포넌트 계층의 중분류 항목들은 다음과 같이 나타낼 수 있다.

| AN : Active Network |
|---|

다. 보조 도메인

능동보안 도메인 분류에서 분류되어지는 5개의 보조도메인 분류를 표시한다.

| 01 : FireWall<br>02 : 인증<br>03 : VPN<br>04 : IDS<br>05 : Active Security |
|---|

라. 소분류

중분류 이하의 기능적 세부성에 따라, 또는 단위 능동 보안 컴포넌트, 관리 컴포넌트, 통합컴포넌트의 3개의 레이어에 따라 하나이상의 하이폰으로 연결된 하위 단계를 가지며 이들은 하이폰(-)으로 연결되며 입자성이 제일 낮은 컴포넌트들을 두 자리 숫자로 표현한다.

3.3.2 컴포넌트 분류 코드 예제

앞 장에서 식별한 능동보안 영역의 컴포넌트들의 코드화를 도식화하고 이러한 컴포넌트 분류 후 항목들에 대하여 그림 3과 같이 분류코드를 배정해 표현할 수 있다. 이러한 분류 코드는 능동 보안 도메인에 대한 저장소에 저장 및 분류 그리고 검색 등에 사용되어 질 수 있으며 각각의 필드는 대분류로 도메인 계층임을 나타내는 "D" 코드를 선두로 능동보안 영역임을 나타내는 AN이 중분류로 표기된다. 다음은 소분류 계층으로 기능적 분류와 세부적인 상세성에 따라 세 자리의 숫자로 표시된 기호들이 하이픈으로 연결된다.

## 3.4 능동 보안 컴포넌트 명세 제안

### 3.4.1 능동보안 명세 개요

능동 보안 컴포넌트는 단독으로 수행되는 작은 응용이라기 보다는 조립되고 커스터마이져되어짐으로써 비즈니스 로직을 수행하는 부품이다. 따라서 오직 정규 포맷을 따르는 인터페이스로 명세화된 컴포넌트만이 활용 가능하다. 현재 활용되고 있는 컴포넌트 명세는 인터페이스 서술을 간과한 채 개략적인 기능적 서술과 사용 환경과 같은 항목으로만 사용자에게 컴포넌트를 선택하도록 한다. 따라서 시스템으로의 통합이나 컴포넌트간의 조립을 위해 정확한 컴포넌트의 식별이 불가능하다. 또한 IDL(Interface Definition Language) 형식의 구문적 명세는 보다 인터페이스 서술의 상세함을 가지더라도 컴포넌트의 행위적인 프로세스가 결여됨으로써 조립을 위한 부가적인 비용 역시 복잡하게 요구된다. 또한 특정 목적을 수행하는 컴포넌트 도메인에 대하여는 부족한 점들이 발견되어질 수 있다. 즉, 시스템으로의 적용시 개발자의 자의적인 해석에 의한 원래 컴포넌트 개발

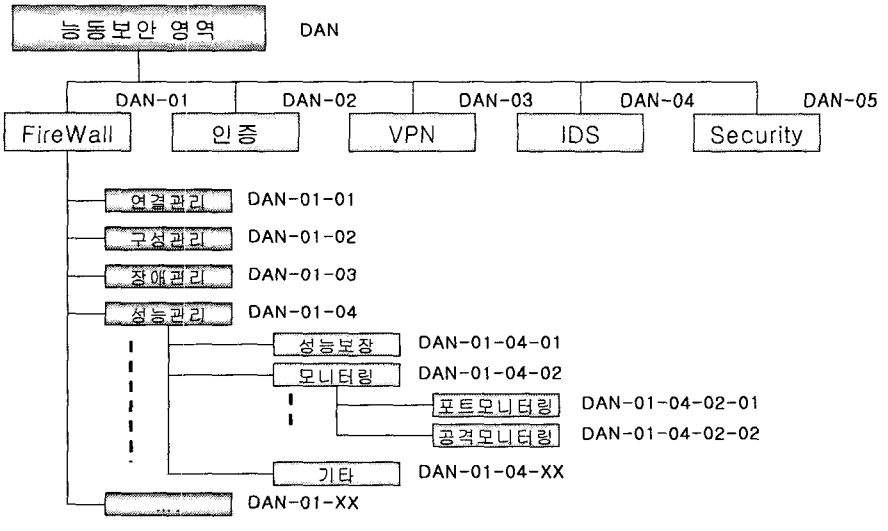



그림 3. 능동보안 컴포넌트 분류코드 사용예

표 2. 컴포넌트 명세에 요구되는 특성

| 항 목 | 의미 |
|---|---|
| Signature | 컴포넌트 외부와 상호작용을 위한 필수 메커니즘의 구문적 관점 |
| Constraints | 개별 signature 요소들의 의미적 명세와 적절한 사용을 위한 제약 |
| Configuration | 사용 시나리오에서 컴포넌트 역할에 따른 인터페이스 패키징(사용 문맥) |
| Quality attributes | 컴포넌트의 비기능적 속성 |

표 3. 컴포넌트 명세 방법

① 컴포넌트 명세를 상세함의 수준에 따라 사용 명세와 개발 명세 두 가지로 정의
② 각 인터페이스 항목의 의미를 명확히 하기 위해 비정규적인 문서를 이용하여 명세 항목마다 '비고'란을 작성하고 pre-/post-condition을 통해 적용 범위를 명확히 정의
③ 컴포넌트 내 구성 요소들 간의 관련성 표현을 위해 순차도(Sequence Diagram)와 상태 전이도(State Transition Diagram)를 작성
④ 컴포넌트 사용의 신뢰성 보장을 위해 사용 시나리오를 작성하고 카테고리의 내에 시나리오 컴포넌트 설정
⑤ 컴포넌트 개발 환경과 컴포넌트 자체의 variant/invariant를 명시
⑥ 컴포넌트 보안성의 정도
⑦ CA 정보
⑧ 네트워크 트레픽 점유도(%)

의도와의 비일치가 발생하며 적절한 인터페이스의 행위적인 의미적 확보가 미흡함에 따라 조립을 위한 쉬운 플러깅 지점을 가질 수 없다. 그러므로 컴포넌트 명세에는 다양한 플러깅을 지원하고 카테고리 되어진 컴포넌트 군으로서 비즈니스 로직을 일관적으로 수행하기 위해 컴포넌트 명세에 표 2와 같은 요소들이 요구된다.

시그네쳐는 외부에서 관찰 가능한 구조적인 부분인 속성과 컴포넌트의 행위적인 능력을 획득, 제공하는 오퍼레이션, 동적인 상호작용을 위한 모델을 형성하는 이벤트로 구성된다. 제약조건은 선/후행 조건의 속성 제한으로 오퍼레이션 의미를 명확히 한다. 구성은 특정한 상호 작용시 몇몇 속성과 오퍼레이션만이 활성화되는 특징에 기반하여 컴포넌트 사용 문맥에 대한 시나리오를 의미하며 품질 속성은 구현, 개발, 유통 등의 비기능적 속성과 보완, 성능, 신뢰성에 대한 요소이다.

### 3.4.2 제안 컴포넌트 명세 방법

#### 가. 개요

본 논문에서는 기존에 제시된 컴포넌트 명세 기법에 새롭게 요구되는 명세 특성들을 포함하여 새로운 컴포넌트 명세 방법을 제안한다. 능동보안 컴포넌트 아키텍쳐에 기반한 특징을 요약하면 표 3과 같다.

#### 나. 컴포넌트 명세 항목

먼저, 컴포넌트의 개요적인 이해와 비즈니스 카테고리 상에서의 위치 파악을 위한 사용 명세로서 본 논문에서는 표 4와 같이 개략 명세 항목을 정의했다.

다음으로, 컴포넌트 개발을 위한 상세 명세로서 응용으로의 전개시 조립을 위한 명확한 의미적인 플러깅 지점을 확보하고 비즈니스 프로세스의 계층적 실현을 위해 표 3.의 특징에 따라 표 4.와 같은 명세 항목을 결정했다. 명세시 각 항목의 비고란을 통해 상세한 부가 설명이 포함된다. 컴포넌트 다이어그램 항목에는 불변성과 가변성, 예외상황으로 인터페이

표 4. 컴포넌트 개략 명세 항목

| 항 목 | 의 미 |
|---|---|
| Code | 컴포넌트 분류 코드 |
| Name | 컴포넌트 식별 이름 |
| Main Goal | (불변의)핵심적인 기능성 |
| Description | 전체적인 개요(기능, 요소, 기대 효과) |
| Related Component | 상호 관계가 있는 컴포넌트 |
| Provider | 컴포넌트 제공자(업체) |
| Version | 컴포넌트 버전 |
| Environment | 호환 가능한 플랫폼(시스템 환경) |
| Security Level | 보안등급 |
| CA | 컴포넌트 인증기관 |
| Traffic | 네트워크 트레픽 점유도 |

스를 명확히 한다. 이용 시나리오 항목은 카테고리 내의 컴포넌트들의 이용 절차를 예시한 것으로 조립을 위한 확신된 가이드라인으로서 이용한다. 품질 요소에는 컴포넌트의 성능과 보완성을 플랫폼 관점에서 준수해야 하는 요소들이 나열된다.

위에서 정의한 능동보안 컴포넌트 명세항목별 설명은 아래와 같으며, 표 5는 능동 보안 컴포넌트 중 하나를 명세한 예이다.

- Code : 능동 보안 컴포넌트 분류코드 배정 방법에 의한 코드를 서술한다.
- Name : 컴포넌트의 식별을 위한 이름을 서술한다.
- Main Goal : 컴포넌트가 수행되어져야 할 기본 목적을 정의한다.
- Description : 컴포넌트의 기능, 필요 요소, 그리고 해당 컴포넌트의 기대 효과등을 서술하여 개발자 또는 사용자가 컴포넌트의 개발 및 획득시 기능을 인지 할 수 있도록 서술한다.
- Related Component : 상호 관계가 있는 컴포넌트를 표시하며 Context Diagram, Component Diagram, Interaction Diagram 그리고 Interface 명세를 도식화함으로서 컴포넌트 내부의 개략 구조와 다른 컴포넌트들과의 관계를 알수 있도록 서술한다.
- Provider : 컴포넌트의 개발사(자) 또는 공급사(자)를 서술함으로서 출처를 명시 및 라이센스를 알 수 있도록 한다.
- Version : 동일 기능의 여러개의 컴포넌트가 존재시 형상관리가 이루어 질 수 있도록 컴포넌트 버전 명세하고 해당 컴포넌트를 획득하여 개발하는 개발자에게는 신버전의 존재를 알 수 있도록 한다 .
- Environment : 개발 환경과 운용 환경 그리고 추가 필요로 하는 운용 정보를 명세함으로서 해당 컴포넌트를 획득하여 사용하는 개발자, 저장소 관리자에게 유용한 정보를 제공한다.
- Security Level : 해당 컴포넌트가 보안을 요구하는 컴포넌트일 경우 해당 보안 등급을 L1에서 L10까지 10개 등급을 명세(외부 보안 인증기관의 공증을 받은 경우 인증기관과 인증 등급을 명세)함으로서 자신이 조합하는 컴포넌트의 보안 능력을 알 수 있도록 한다.
- CA : 컴포넌트를 저장소에 저장 및 획득 시 이를 인증하여 준 기관 명세함으로서 공증화 할 수 있는 자료를 제공함으로서 신뢰성을 확인할 수 있도록 한다.
- Traffic : 해당 컴포넌트가 사용되어질 때 기본적인 네트웍 사용량이 아닌 컴포넌트에 의해 발생되어지는 네트웍의 트레픽의 점유도를 서술함으로서 개발자는 해당 점유도에 맞도록 개발하며 획득자는 해당 컴포넌트의 네트웍 트레픽 점유도를 알 수 있도록 표현한다.

## 4. 결 론

본 연구에서는 인터넷의 보급으로 개발자 중심의 S/W 개발 방식에서 사용자 중심의 개발환경으로 변화되는 현 시점에서 빠르게 바뀌는 환경에 적절하게 대응할 수 있는 컴포넌트 기반의 표준화된 능동보안 컴포넌트 설계기법 정립을 위하여 컴포넌트 아키텍쳐를 이용한 설계방안을 제시하고 능동 보안 도메인을 개발하며 능동 보안 컴포넌트를 개발하기 위한 설계 명세를 제안한다.

분산 시스템들의 증가와 인터넷의 확산으로 인하여 네트워크를 통한 공격의 가능성은 점점 늘어나고 있다. 잠재적인 공격의 위협으로부터 시스템을 보호하기 위해서 많은 조직들은 침입탐지시스템, 침입차단시스템 등의 정보 보호 시스템들을 배치하고 있다. 그러나 기존의 보안 장치들은 지역 네트워크 경계를 넘어선 탐지가 불가능하고, 네트워크 차원의 효율적이고 적극적인 대응이 불가능하다. 설사 다른 영역의 공격을 탐지했을 지라도 원격으로 다른 네트워크에 대응을 할 수 있는 능력이 없다. 또한 새로운 공격 패턴이나 보안 정책 등의 변화에 적응이 어렵다. 이러한 문제점들을 해결하기 위해서는 공격에 대해 능동적으로 대응이 가능하며, 보안 시스템들을 상호 결합하여 운용하고, 보안 정책 및 기술의 수용이 용이한 보안 구조가 필요하다. 이런 요구 조건들을 충족시키기 위해 DARPA에서는 능동 네트워크를 제안하였다. 그러나 능동 네트워크는 동적이고 유연한 본성으로 인해 그 자체로도 심각한 보안 위협을 가지고 있다. 따라서 기존 네트워크의 보안 문제점을 해결하고 능동 네트워크의 보안 위협에 대한 자체적인 방어 능력을 가지고 있는 능동 보안 관리를 위한 능동 네트워크 구조에 대한 연구가 필요하다. 본 연구에서는 전반부에서 능동보안에 대하여 분석하였으며 이 자료를 토대로 능동보안 도메인을 분석하였고 능동 보안 컴포넌트를 개발 할 수 있는 레퍼런스 모델들을

표 5. 컴포넌트 명세 예제

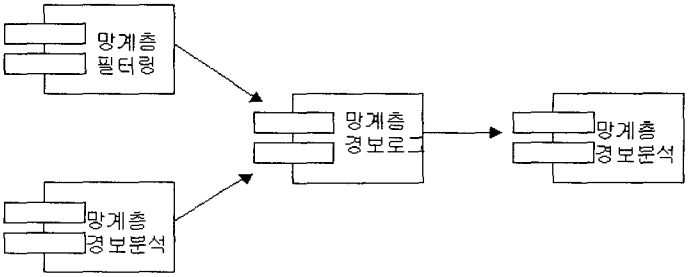
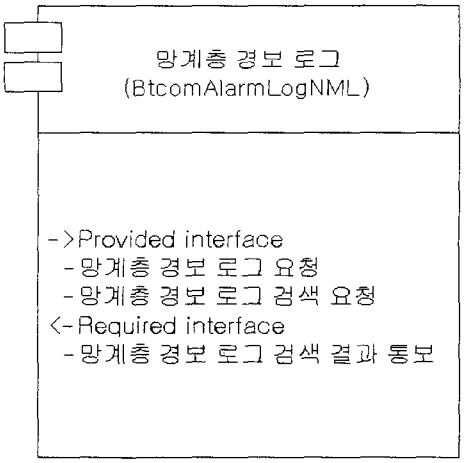
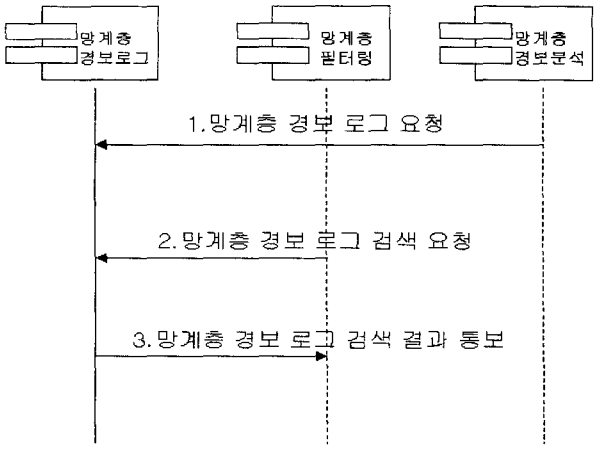
| Code | DAN-01-03-01-01 |
|---|---|
| Name | 망계층 경보 로그 |
| Main Goal | 장애 데이터의 RAW 형태 로그 생성 |
| Description | 망계층 경보로그는 Firewall 기능을 수행 하기 위한 단위 비즈니스 로직으로서 망계층에서 필터링을 통하여 장애에 관련된 데이터(장애 발생자원, 장애발생 유형 및 원인)를 RAW데이터 형태로 관리하기 위한 컴포넌트이다. |
| Component Context Diagram |  |
| Component Diagram |  |
| Component Interaction Diagram |  |

표 5. 계속

| | |
|---|---|
| Component Interface | - 망계층 경보 로그 요청<br>Provided Interface 망계층 경보 로그 요청 [to] 망계층필터링<br>Description : 망계층 필터링 컴포넌트가 장애관리기능 수행을 위한 정보를 기록<br>[Preconditions : 기본적으로 활성 상태]<br>[Postconditions : 기록된 정보에 대한 주기적인 갱신 및 삭제]<br>Input : 관리수행 컴포넌트 이름, 수행시간, 장애 관련 정보<br>Output : None<br>- 망계층 경보 로그 검색 요청<br>Provided Interface 망계층 경보 로그 검색 요청 [to]<br>망계층 경보분석<br>Description : 망 계층 장애 조정 기능의 망계층 경보분석<br>컴포넌트가 장애의 근본 분석을 위한 망 계층<br>경보로그 컴포넌트에서 제공하는 인터페이스<br>[Preconditions : 장애발생 자원 및 장애발생 원인에 대한 정보를 가지고 있어야 함]<br>[Postconditions : 요약 감시 결과를 저장 및 로고 제어 기능 수행]<br>Input : 장애 발생 자원의 ID, 장애의 유형 및 원인<br>Output : 장애 발생 자원의 ID, 장애의 유형 및 원인 |
| Provider | Daejeon University & ETRI |
| Version | 0.1 |
| Environment | EJB 2.0 Spec. , Weblogic, JDBC, mSQL |
| Security Level | L10 |
| CA | ETRI |
| Traffic | 1% 미만 |

연구하였으며, 이를 바탕으로 능동보안 컴포넌트에 대하여 연구하였다.

## 참 고 문 헌


[1] 네트워크보안연구부, 차세대 인터넷을 위한 능동보안 기술 백서, ETRI, 2001.

[2] http://wwwdarpa.mil

[3] http://www.kanf.or.kr

[4] Dan Schneckenberg, Kelly Djahandari and Dan Sterne "Infrastructure for Intrusion Detection and Reponse", DISCEX 2000, Jan. 25-27, 2000.

[5] D.S. Alexander et al., "Secure Active Network Encapsulation Protocol(ANEP)", http://www.cis.upenn.edu/.

[6] K. Psounis, "Active Network: Application, Security, Safety, and Architectures", IEEE Communication Surveys, 1999.

[7] James Carey, Brent Carlson, Tim Graser, "San-Francisco Design Patterns", Addison Wesley, 2000.

[8] Douglas Schmidt, Michal Stal, Hans Rohnert, Frank Buschmann, "PATTERN-ORIENTED Software Architecture" Wiley, 1999.

[9] John Cheesman, John Daniels, "UML Components", Addison Wesley, 2000.

[10] Clemens Szyperski, "Component Software", Addison Wesley, 1997.

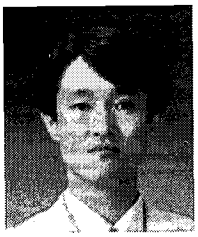
[11] 김상영, 황선명, 나중찬, "능동보안 컴포넌트 개발에 관한 연구," 정보처리학회 추계학술발표 논문집, 제9권 2호, pp. 1929-1932, 2002.

[12] 김상영, 김재웅, 황선명, "능동보안 컴포넌트에 관한 연구," 정보과학회 소프트웨어공학회지, 제15권 4호, pp. 3-11, 2002.

[13] Ruben Prieto Diaz, Implementing faceted classification for Software Reuse, Communications of The ACM, Vol. 34, No 5. 1991.

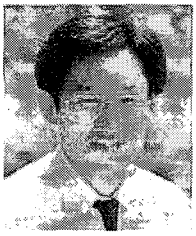
김 상 영

1999년 대전대학교 컴퓨터공학과 졸업(학사)
2001년 대전대학교 대학원 컴퓨터공학과(석사)
2001년~현재 대전대학교 대학원 컴퓨터공학과 박사과정
관심분야 : 소프트웨어 테스팅, 소프트웨어 개발환경



김 재 웅

1983년 중앙대학교 전자계산학과 졸업(학사)
1988년 중앙대학교 대학원 전자계산학과(석사)
2000년 대전대학교 대학원 컴퓨터공학과(박사)
2001년~현재 공주대학교 멀티미디어정보·영상공학부 부교수
관심분야 : 소프트웨어공학, 소프트웨어 개발방법론



황 선 명

1982년 중앙대학교 전자계산학과 졸업(학사)
1984년 중앙대학교 대학원 전자계산학과(석사)
1987년 중앙대학교 대학원 전자계산학과(박사)
1988년 독일 BONN대학 Post Doctor
2000년~현재 한국S/W프로세스 심사인협회(KSPICE) 이사
1997년~현재 ISO/IEC JTC7/WG10 한국운영위원
1998년~현재 한국정보통신기술협회 TTA 특별위원
1989년~현재 대전대학교 컴퓨터공학과 교수
관심분야 : 소프트웨어 프로세스 모델, 품질 메트릭스, 소프트웨어공학 표준화, 컴포넌트 품질측정, 테스팅 방법론 등