SAML 기반 VPN 구성을 위한 SSO 관리구조

강명수 홍충선

◈ 목 차 ◈

1. 서 론

인터넷을 통한 네트워크 운용 및 전자상거래 등의 규모가 커짐에 따라, 트랜잭션의 안정성, 망 자원의 보호, 사용자의 금융정보보호 등 정보보안 및 보호의 문제가 중요한 이슈로 제기되고 있다. 네트워크보안 및 전자거래보안으로 주로 연구되고 있는 정보보호 기술로는 password기반, PKI(Public Key Infrastructure)기반[1], SSL(Secure Sockets Layer) 기반의 보안서비스로부터 XML(eXtensible Markup Language)에서 출발한 ebXML, Web Service기반으로 하는 XML정보보호기술 기반의 전자상거래 지원 보안서비스로까지 발전하고 있다. 이러한 기술개발 동향에서도 알 수 있듯이 인터넷 전자문서가 차츰 XML기반으로 표준화되어 전자결제, 전자계약 등 전자상거래 서비스의 XML화가 급속히 진행되고 있으며 ebXML(e-business XML), 웹서비스 등 국제 전자상거래 표준이 XML기반으로 이루어지고 있다.

기존의 SSL 기반으로 데이터를 전송할 때는 데이터 전체에 대해 암호화를 수행하기 때문에 데이터 자체에 대한 기밀성을 보장할 순 있었으나 데이터의 일부분만 암호화할 경우엔 부적절하였다. 또한 PKI 기반의 인증 보안서비스는 구조 및 코드가 복잡하여 구현할 때 많은 비용과 노력이 요구되는 문제점을 안고 있었다. 이에 반해 XML Encryption은 이진데이터를 암호화하는 연산과 XML 데이터를 암호화하는 연산으로 나뉘어 XML데이터를 암호화하는 연산은 데이터 중 일부분 또는 전체를 암호화하여, 중간에 경유하게 되는 제3자에게 특정 정보를 노출시키지 않으면서 최종 수신자에게 전달할 수 있다. 또한 XML기반 기술은 데이터의 일부분 또는 전체를 암호화하여 최종 수신자에게 전달할 수 있도록 하면서 복잡한 PKI방식에 비해 단순한 구조로 이루어져 시스템 간 손쉽게 데이터를 교환할 수 있다.

본 논문에서는 이러한 장점을 가진 SAML을 VPN에 적용시켜 사용자와 관리자 간 안전하게 인증 정보를 주고받을 수 있는 인증 시스템을 설계하였다.

2장에서는 관련연구로서 SAML의 구조 및 인증 시나리오 중심으로 기술하고 Single Sign On, LDAP 등의 개념 중심으로 정리한다. 3장에서는 SSO 인증 시스템 모델에 대해서 정의한다. 또한 VPN에서의 SSO 과정과 SAML assertion, artifact를 이용해 SSO 서비스가 제공되는 과정을 제시한다. 4장에서는 시뮬레이션을 통해 성능을 평가하며 5장에서 구현 및 성능평가 결과에 대하여 논의한다.

* 경희대학교 컴퓨터공학과

2. 관련연구

2.1 SAML

2.1.1 SAML 개요 및 구조

SAML[2] OASIS(Organization for the Advancement of Structured Information Standards)에서 개발된 XML(eXtensible Markup Language)프레임워크로, 그림 1에서와 같이 플랫폼의 거래 파트너들이 인증 정보, 권한 부여 정보, 그리고 프로파일 정보를 안전하게 교환할 수 있도록 설계된 것이다. 이것은 기업 내부 또는 기업 간의 Single Sign-On을 제공하고, 기반 보안 infrastructure에 종속되지 않는다. SAML은 이전의 두 보안 기술, 즉 S2ML (Security Services Markup Language)과 Auth-XML (Authorization XML) 에서 파생된 기술로, SAML이 출현하기 전에는 기업은 독자적인 인터페이스, 특정한 Single Sign-On 제품 또는 디렉터리 기반 제품 중 하나를 사용해Single Sign-On을 해결해야 했다.

SAML의 장점은

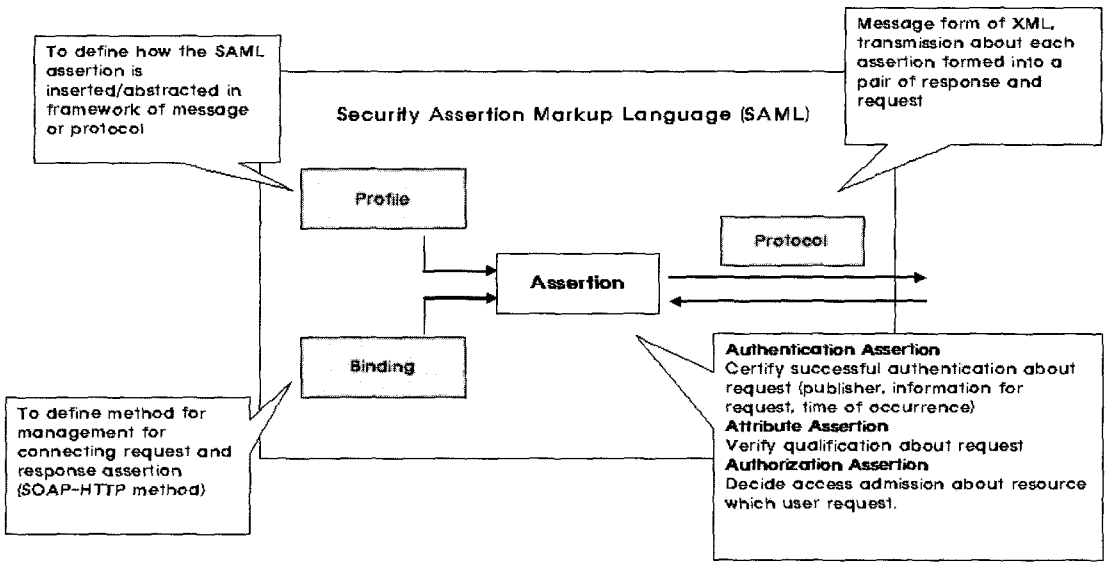
- XML 기반으로 XML에서 제공하는 장점을 사용 가능하다.
- SAML을 사용해서 한번 인증정보를 입력하면 다른 다양

한 영역에서도 인증을 받을 수 있는 Single Sign-On이 가능

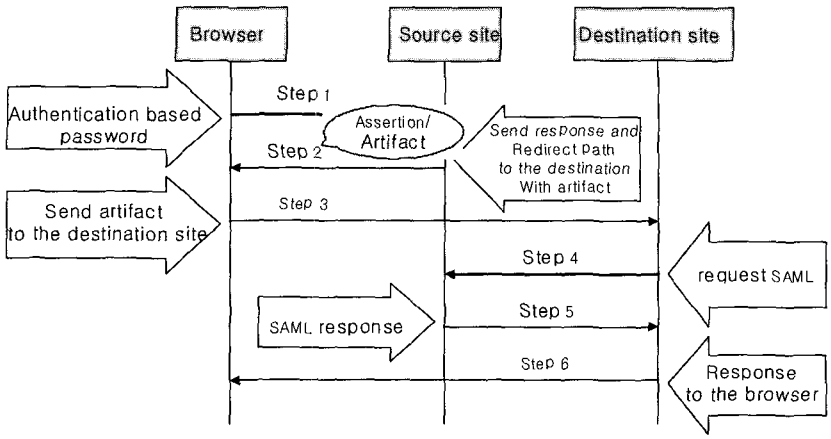
- SOAP(Simple Object Access Protocol)이나 ebXML등의 프로토콜과 함께 사용 가능하다는 점 등을 들 수 있다.

플랫폼 독립적인 SAML은 Assertion, Profile, Binding, Protocol 로 구성되어 있다(그림 1).

- Assertion : Assertion은 아이덴티티 기관에서 최종 사용자(사람이나 기계)에 대해 만든 구문(Statement)을 말한다. Assertion은 '특정 사용자가 해당 애플리케이션 웹사이트에 대한 접근을 허가 받았는가'와 같은 요청에 응답한다. Assertion에는 권한 부여(authorization)와 인증(Authentication), 속성(Attribute)등 세 가지 유형이 있다.
- Protocol : SAML Protocol은 요청과 응답의



(그림 1) SAML 구조



(그림 2) Artifact기반의 SAML Pull 모델

포맷을 결정한다.

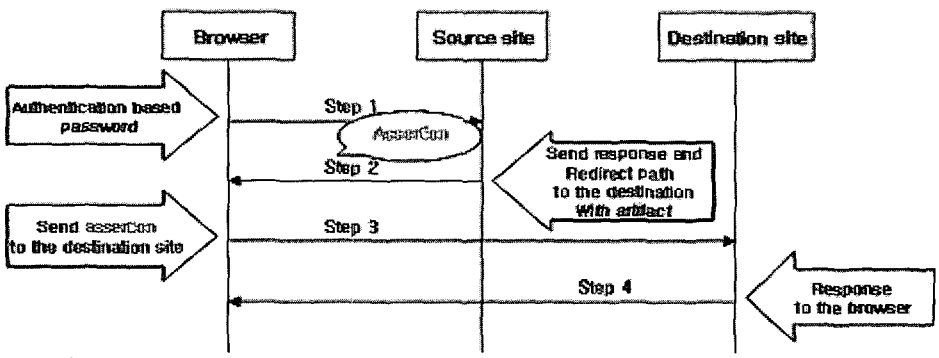
- Binding : Binding은 SAML요청과 응답을 전송하는 방식이다. SAML에 대해 가장 널리 사용되는 전송 프로토콜은 HTTP를 통한 SOAP이지만, OASIS에서는SAML용으로 순수한 HTTP 전송수단에 대한 작업을 진행 중에 있다.
- Profile : SAML assertion이 메시지 안에서 발견될 수 있는 장소와 같은 것들을 설명하고 있다.

2.1.2 SAML을 사용한 인증 시나리오

SAML 생성은 사용자가 최초의 인증을 받을 때 소스 사이트를 통하여 만들어지며 이는 토큰과 같은 형식으로 생성된다. SAML 인증 모델은 크게 두 가지로 나뉘어 조금 다르게 동작된다. Pull 모델과 Push 모델이 그것이다.

SAML 인증 Pull 모델(그림 2)에서 end host는 소스 사이트에 assertion을 요청한다. 소스사이트는 인증/승인 과정을 거쳐 artifact를 생성하고 사용자에게 전송한다. Artifact는 assertion의 특별한 형태이다. Artifact는 소스사이트에 저장되어 있는 assertion을 참조하는 포인터와 같다. 웹 사용자가 목적지 웹사이트에 자원을 요청하면 소스 웹사이트는 사용자에게 artifact를 전달함과 동시에 목적지 웹사이트로 경로 재설정을 수행한다. 목적지 웹사이트는 수신한 artifact를 소스 웹사이트에 전달하고 해당 assertion을 전달받는다. 마지막으로 목적지 웹사이트에서는 수신한 assertion을 분석하고 적합한 요청일 경우 SOAP이나 HTTP같은 전송 프로토콜을 사용해 웹 사용자에게 요청 자원을 전달한다[2].

SAML Push 모델에서는(그림 3) 사용자가 소스



(그림 3) Assertion기반의 SAML Push 모델

웹사이트에 ID/Password기반으로 인증 요청을 하고 소스 웹사이트에서 인증/승인 후 인증 사용자에 해당하는 Assertion을 생성한다. Pull모델에서와 같이 사용자는 후에 소스 사이트에 목적지 웹사이트의 자원을 요청하고 소스 사이트는 요청자에게 assertion을 전달함과 동시에 목적지 웹사이트로 경로 재설정을 수행하게 된다. 목적지 웹사이트는 요청자로부터 assertion을 전달받은 후에 소스 웹사이트와의 별도의 프로토콜 통신인증 없이 사용자를 인증하게 된다.

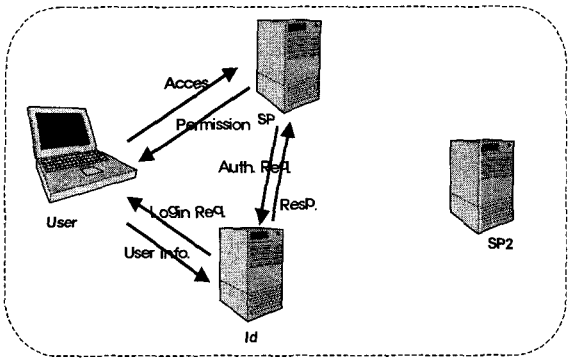
```
-<saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
 AssertionID="cT_S_T-vKMwidT8_Pzkke8UkC68." IssueInstant="2004-02-25T16:31:03Z"
 Issuer="http://aaremote.entegrity.com" MajorVersion="1" MinorVersion="1"
 xmlns:ds="http://www.w3.org/2000/09/xmldsig#">

 <saml:Conditions NotBefore="2004-02-25T16:26:03Z"
 NotOnOrAfter="2004-02-25T16:36:03Z" />
-<saml:AuthenticationStatement AuthenticationInstant="2004-02-25T16:30:58Z"
 AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
+<saml:Subject>
-<saml:SubjectConfirmation>
 <saml:ConfirmationMethod>
 urn:oasis:names:tc:SAML:1.0:cm:bearer
 </saml:ConfirmationMethod>
 </saml:SubjectConfirmation>
 </saml:Subject>
 <saml:SubjectLocality IPAddress="192.168.4.1" />
 </saml:AuthenticationStatement>
-<saml:AttributeStatement>
+<saml:Subject>
+<saml:SubjectConfirmation>
 </saml:SubjectConfirmation>
 </saml:Subject>
+<saml:Attribute AttributeName="Assurance-Level"
 AttributeNamespace="http://www.oasis-open.org/RSA2004/attributes">
 </saml:Attribute>
+<saml:Attribute AttributeName="E-mail"
 AttributeNamespace="http://www.oasis-open.org/RSA2004/attributes">
 </saml:Attribute>
+<saml:Attribute AttributeName="Member-Level"
 AttributeNamespace="http://www.oasis-open.org/RSA2004/attributes">
 </saml:Attribute>
+<saml:Attribute AttributeName="common-Name"
 AttributeNamespace="http://www.oasis-open.org/RSA2004/attributes">
 </saml:Attribute>
 </saml:AttributeStatement>
 </saml:Assertion>
```

위는 SAML기반의 인증과정을 거쳐 생성된 Assertion의 형식을 나타낸 것으로 이와 같이 XML 기반의 Assertion은사용자 요청을 인증/승인할 수 있는 충분한 정보를 담고 있다.

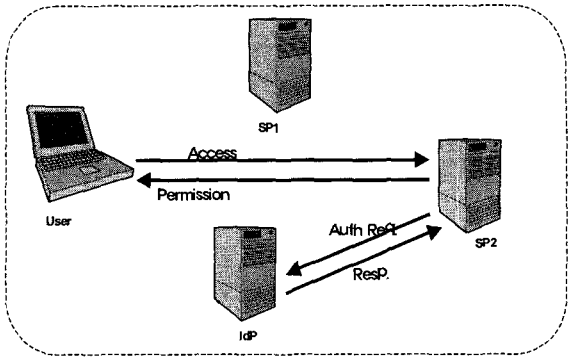
2.2 Single Sign On Scenario

하나의 아이디로 여러 사이트를 이용할 수 있는 시스템으로 여러 개의 사이트를 운영하는 대기업이나 인터넷 관련 기업들이 각각의 ID를 하나로 통합할 필요성이 대두됨에 따라 개발된 방식이다[3]. 1997년 IBM이 개발하였고 우리나라에는 2000년 코리아닷컴이 처음 도입한 이후 삼성전자와 SK가 도입하며 활성화되어 애니패스와 롯데타운 등 다양한 사이트와 많은 솔루션 공급업체 등에서도 사용하기 시작했다. 개인은 사이트에 접속하기 위해 아이디와 패스워드는 물론 개인 신상정보도 일일이 입력해야 하는 불편함을 한 번의 작업으로 해소시켜주고, 기업은 회원을 통합적으로 관리할 수 있어 마케팅 효과를 극대화시킬 수 있는 장점이 있다.

그림 4와 그림 5는 single sign on(SSO)의 기본적인 동작 과정을 나타낸 것이다. 그림에서 SP는 service provider를 나타내고 IdP는 Id provider, 즉 인증을 담당하는 기관이다. 인증되는 순서는 다음과 같이 이루어진다[4].



(그림 4) Single Sign On 인증 모델 (1)



(그림 5) Single Sign On 인증 모델 (2)

- 사용자가 sp1에 URL로 접속한다.
- sp1은 IdP에게 사용자의 정보를 인증해줄 것을 요청한다.
- IdP는 사용자에게 로그인화면으로 redirect하여 인증을 요구한다.
- 사용자는 아이디와 패스워드를 입력한다.
- IdP는 사용자 정보를 토대로 인증을 하고 sp1에 인증서를 보내준다.
- sp1은 받은 인증서를 토대로 요청받은 사용자 인증이 유효함을 확인한다.
- sp1은 사용자가 인증된 사용자로 판명되면 인증이 완료된 브라우저를 넘겨준다.

다시 사용자는 이번에는 sp2의 서비스를 이용하려 한다. 이때는 다음과 같은 순서로 인증이 이루어진다[5].

- 사용자가 sp2에 접속을 요청한다.
- sp2는 사용자의 정보를 IdP에 요청한다.
- IdP는 sp2로부터 요청을 받고 세션 값을 바탕으로 인증서를 검사하고 사용자가 이전에 sp1에 로그인했었던 사용자임을 알아낸다. IdP는 sp2에 유효한 인증자임을 알려준다.
- sp2는 IdP로부터 유효한 인증 response를 받고 사용자가 이전 sp1에서 로그인했었던 인증된 사용자임을 확인한다.
- sp2는 사용자에게 인증이 완료된 브라우저는 넘겨주고 사용자는 서비스를 이용하게 된다.

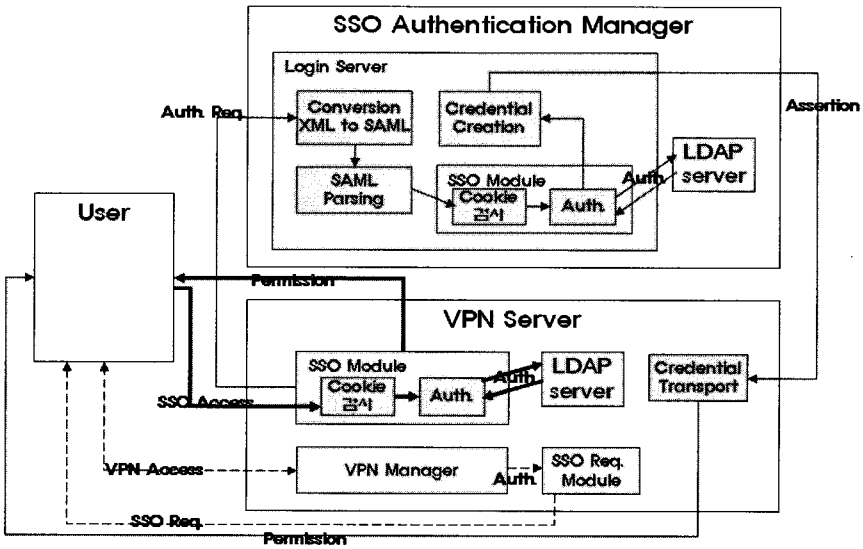
3. 제안구조

본 장에서는 VPN에서 사용자와 OSS간 안전하게 정보를 교환하기 위해서 SAML을 사용해 Single Sign On을 제공하는 인증시스템의 구조와 각 모듈의 역할에 대해서 제안사항을 중심으로 설명한다. VPN 터널링을 통해 사용자와 OSS간 정보를 교환한다고 가정하고 VPN 관리 시스템의 login server에서 전송받은 데이터를 SAML로 변환시켜 인증을 수행하고 assertion을 생성하는 모듈과 이후 VPN server에서 Single Sign On을 제공하는 모듈을 제시한다.

3.1 SAML 기반의 데이터 전송 내부 모듈

VPN 간 single sign on을 제공하기 위해 SAML 기반으로 인증을 수행하는 시스템에서 VPN server와 VPN client, VPN 관리시스템 간 정보가 전달되는 내부 구조는 다음과 같다[7]. 그림 6을 보면 각 시스템마다 가지는 모듈의 형태가 다르고 사용자(user), VPN 관리시스템, VPN server는 서로 정보를 교환한다.

사용자URL, 아이디/패스워드, assertion/cookie등의 값을 주고받는다. URL은 처음 VPN server에 접속할 때, 그리고 아이디/패스워드는 추후에 VPN 관리시스템으로부터 로그인 요청을 받았을 때 사용된다. 마지막으로 assertion/cookie는 타 server들과



(그림 6) SAML 기반의 데이터 전송 내부구조

연동 시 생기는 정보들이다.

VPN manager는 사용자가 VPN에 등록되어 있는 사용자인지를 검사하는 모듈로써 VPN에 등록된 사용자라면 VPN 매니저는 SSO 요청모듈로 SSO 요청 메시지를 보낸다.

SSO인증관리자(Authentication Manager)는 Login server와 LDAP 서버를 가진다. Login server는 SAML 변환모듈, SAML parsing 모듈, SSO 모듈, assertion생성모듈로 구성되어 있다. SAML 변환모듈은 외부로부터 들어온 XML기반의 데이터를 SAML으로 변환시켜주는 역할을 담당한다. SAML 파싱(parsing) 모듈은 변환된 SAML을 일반 프로그램 언어를 사용하여 인증할 수 있도록 문법검사를 수행한다. SSO 모듈은 쿠키(cookie) 검사 모듈과 인증 모듈로 나뉘는데 쿠키 검사 모듈은 쿠키를 통해 파싱된 데이터(즉, 사용자 정보)가 검증된 사용자인지를 가려내는 역할을 담당한다. 또는 사용자가 이전에 로그인 과정을 거쳐 세션을 가지고 있는지 검사한다. 인증 모듈은 최종적으로 사용자 정보를 가지고 인증을 수행하는 부분이다.

VPN server는 SSO 모듈과 LDAP 서버, assertion 전송 모듈을 가지고 있다[8][9][10]. SSO 모듈은 VPN 관리시스템의 SSO 모듈과 같이 쿠키 검사 모듈과 인증 모듈로 나뉜다. 쿠키 검사 모듈은 세션 검사를 통해 접속을 요청한 사용자가 이전에 로그인했었던 사용자인지 가려낼 수 있다. assertion 전송 모듈은 VPN 관리시스템에서 생성된 assertion을 받아서 사용자에게 전송하는 역할을 담당한다.

그림 6의 구조를 토대로 사용자 인증이 이루어지는 순서는 다음과 같다. 먼저 사용자가 처음 시스템에 로그인할 때 순서는 아래와 같다.

- 접속 요청 - 사용자가 이전에 로그인했었는지 여부를 알리기 위해 처음으로 VPN server에 접속을 요청한다.
- 인증 요청 - VPN server가 login server에게 사용자를 인증해줄 것을 요청한다. login server는 사용자에게 아이디와 패스워드 정보를 요구한다.
- 인증 - 사용자가 전송한 정보를 토대로 login server와 LDAP간 연동을 통해 인증을 수행하고 assertion을 생성한다.
- 인증서 전달 - 생성된 assertion을 VPN server에게 전송한다.
- 인증 성공 - VPN server가 생성된 assertion과

로그인 성공 browser를 함께 사용자에게 전송한다. 사용자는 성공 메시지를 받고 OSS와 통신이 가능하다.

로그인 후 다른 시스템에 인증과정 없이 로그인할 때 순서는 아래와 같다.

- 접속 요청 - 사용자가 다른 VPN server에 접속한다. 이전에 로그인하였으므로 로그인 세션을 갖고 있다.
- 인증 - VPN server는 세션을 검사하고 assertion을 요청하고 login server와 공유하는 키와 assertion을 이용해서 인증을 수행한다.
- 인증 성공 - 사용자 정보가 인증되면 사용자에게 인증되었음을 알리고 사용자는 OSS와 통신하게 된다.

3.2 SAML 기반의 VPN 관리 시스템

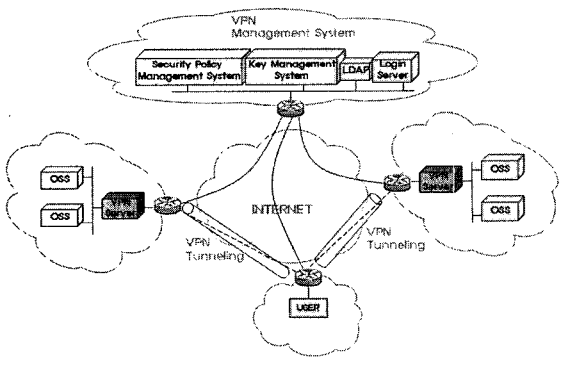
그림 7은 VPN이 있는 네트워크에서 사용자 인증 구조를 나타낸 것이다. 서로 다른 네트워크가 백본망을 거쳐 연결되어 있고 사용자(user)와 OSS 사이는 VPN 터널링을 통해 인증을 거쳐 정보를 주고받는다.

사용자가 특정 OSS에 접속하려면 VPN server에서 인증을 거치고 각 VPN server는 VPN 관리시스템(VPN Management System)에서 통합적으로 관리된다. VPN 관리시스템은 공개키 관리, VPN server 관리 등 사용자가 VPN 망에서 원격 OSS에 접근하고자 할 때 인증에 대한 통합적인 관리를 주관한다[9]. 본 연구에서 제안된 인증시스템 (그림 7)은 VPN이 있는 네트워크에서 사용자와 OSS간 VPN 인증을 통해 정보를 주고받고자 할 때 하나의 아이디로 여러 다른 VPN server에 별도의 로그인 없이도 인증을 가능하도록 하였다.

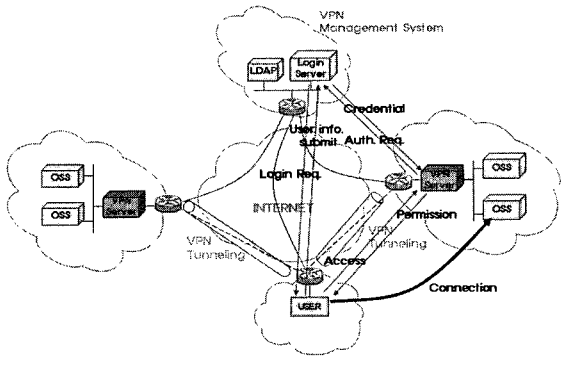
3.2.1 VPN에서 Single Sign On 과정

본 연구의 시나리오는 사용자가 다른 망의 OSS와 정보교환을 하기 위해 접속을 요청하는 것으로 시작된다. VPN에서 single sign on 인증 시스템의 시나리오는 그림 8과 같다.

- 사용자가 특정 OSS에 접근하기 위해 VPN server에 인증을 하려한다. 이 때 사용자는 VPN tunneling을 통해서 망에 접근하고 VPN server에 접속한다.
- VPN server는 접근을 요청한 사용자가 유효한 사용자인지를 알기 위해VPN 관리시스템의 login server에 사용자 인증을 요청한다.
- login server는 인증을 요청받고 해당 사용자에게 로그인 폼을 전송하여 인증을 요구한다.
- 사용자는 아이디와 패스워드를 입력한 폼을 login server에게 전송한다.
- login server는 사용자로부터 아이디와 패스워드 정보를 전송받아 데이터베이스 검색을 통해 유효한 사용자인지 여부를 검사한다. 이 때 데이터베이스는 LDAP 기반으로 LDAP에는 아이디와 패스워드 같은 사용자의 기본정보와 기



(그림 7) VPN에서 인증시스템 구조



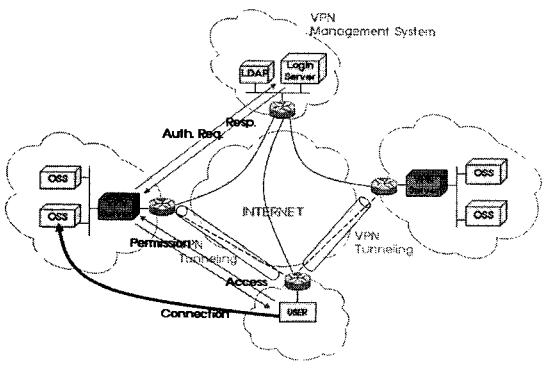
(그림 8) VPN에서 Single Sign On 과정 (1)

타 인적사항이 저장되어 있다. 사용자가 유효하다면 login server는 인증이 되었음을 VPN server에게 알린다.

- VPN server는 login server로부터 유효한 사용자임이 확인된 응답을 받고 사용자에게 인증이 되었음을 알리는 페이지를 전송한다.
- 사용자는 VPN server 인증에 성공하였으므로 VPN 터널링을 통해 OSS 에 접근하여 정보를 교환할 수 있다.

그림 8의 과정을 바탕으로 사용자는 VPN server에 인증 권한을 받아 OSS에 접근하는데 성공하였다. 이제 추후에 사용자가 이전의 인증 받았던 VPN server와는 다른 서버가 위치한 네트워크에 접근하여 또 다른 OSS와 정보를 교환하려고 한다. 이 때 사용자는 별도의 인증을 거치지 않고도VPN server에게서 인증 권한을 받을 수 있다. 그림 9에서 사용자가 다른 망의 OSS와 정보를 교환할 때 VPN server에 인증 받는 순서를 볼 수 있다.

- 사용자가 또 다른 특정 OSS에 접근하기 위해 VPN server에 접속한다. 이 때 사용자는 VPN tunneling을 통해서 망에 접근하고 VPN server에 접속한다. 그림 9에서 사용자는 이미 인증에 성공하였고 VPN server로부터 사용자임을 알려주는 인증서(assertion)를 받았다. 때문에 VPN server로 assertion과 함께 접속을 요청한다.



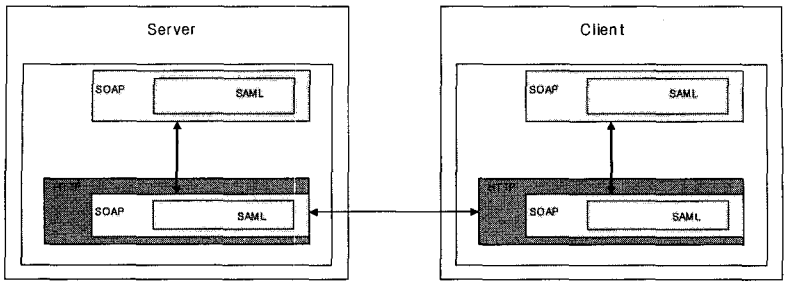
(그림 9) VPN에서 Single Sign On 과정 (2)

- VPN server는 접근을 요청한 사용자가 유효한 사용자인지를 알기 위해 VPN 관리시스템의 login server에 사용자 인증을 요청한다.
- 그림 9에서 login server는 VPN server로부터 사용자 인증 요청만을 받아 사용자에게 로그인 폼을 전송했었다. 그러나 사용자가 assertion을 실어 접속을 요청하였기 때문에 login server는 assertion을 토대로 사용자가 이전에 로그인 과정을 이미 거쳤다는 것을 알 수 있다. login server는 별도의 인증 과정 없이 VPN server에게 인증되었음을 알린다.
- VPN server는 login server로부터 유효한 사용자임이 확인된 응답을 받고 사용자에게 인증이 되었음을 알리는 페이지를 전송한다.
- 사용자는 VPN server 인증에 성공하였으므로 VPN 터널링을 통해 OSS에 접근하여 정보를 교환할 수 있다.

VPN server와 client간에 일반적인 single sign on 과정은 그림 10과 같은 순서로 이루어진다. 본 연구에서는 제안된 VPN server/client 간에 single sign on 인증이 이루어질 때 SAML 기반으로 생성된 assertion을 인증서로 사용하여 정보교환이 이루어지도록 하였다. server와 client 간 SAML 데이터 전송 시에는 SOAP를 사용하여 주고받는다[5].

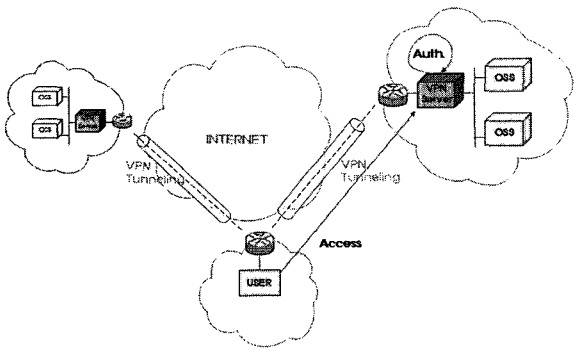
3.2.2 VPN에서 assertion을 이용한 Single Sign On 과정

SAML assertion을 사용해 server와 client간 single sign on이 이루어지는 절차가 다음의 그림들에서 설명될 수 있다. server와 client사이에 인증서가 오고가는 형태는 두 가지로 나뉜다. 한 가지는 assertion을 생성하여 도큐먼트 형태로 주고받는 것이고 다른 한 가지는 artifact라는 작은 데이터를 생성해서 포인터처럼 넘겨주는 방식이다. 우선 첫 번째로, assertion의 형태로 데이터 전송이 이루어지는 과정이 그림 11과 12에서 볼 수 있다. 그림 11은 사용자가OSS와 정보교환을 위해 VPN server에 처음으로 인증을 하는 부분이다.

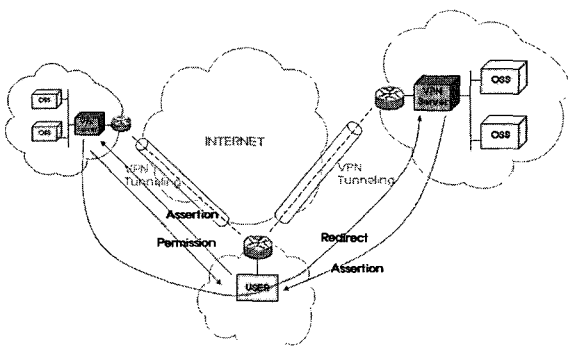


(그림 10) Server 와 Client 간 SAML 정보전달 방식

- 사용자가 다른 망에 있는 특정 OSS에 접근하기 위해 최초로 해당 망 내 VPN server에 접속한다. 이 때 접속 형태는 URL이다.
- VPN server는 VPN 관리시스템에 사용자 유효성 여부를 요청하고 VPN 관리시스템 내 login server는 사용자에게 로그인을 요청한다. 로그인 정보를 넘겨받은 login server는 인증을 거쳐 SAML assertion을 생성한다. 이 때 생성된 assertion은 XML형태의 도큐먼트이다. login server는 이를 데이터베이스에 저장하거나 사용자에게 넘겨준다. 사용자는 인증에 성공하고 OSS와 터널링을 통해 안전하게 정보를 교환할 수 있다. 인증에 성공한 후 사용자는 다시 또 다른 망의 OSS에 접근하기 위해 해당 망 내 VPN server에 접속을 시도한다. 그림 11을 보면, 왼쪽에 있는 부분이 새롭게 접근하려는 네트워크이고 오른쪽의 네트워크가 이전에 로그인했었던 망이다.
- 사용자가 다른 OSS에 접근하기 위해 해당 망 내에 있는 VPN server에 접속을 요청한다. 이 때 접속 형태는 URL이다.
- VPN server는 동일한 VPN 관리시스템의 통제를 받는 VPN server들의 리스트를 사용자에게 보여준다.
- 사용자는 이 중 이전에 로그인에 성공했던 VPN server를 select한다. VPN server는 select된 폼 정보를 사용자로부터 받는다.
- 왼쪽 네트워크 내 VPN server는 사용자에게 select된 VPN server로 redirect 메시지를 보낸다. 사용자에게 보이는 브라우저는 이전에 로그인 과정을 거쳤던 VPN server의 그것이다.
- 오른쪽 네트워크 내 VPN server가 사용자의 assertion을 왼쪽의 VPN server에게 전달해주도록 페이지를 요청한다. 사용자는 응답함으로써 이전에 생성된 사용자의 assertion을 전송받는다.
- 사용자가 오른쪽의 VPN server로부터 전송받은 assertion을 왼쪽의 VPN server에게 넘겨준다.



(그림 11) SAML assertion 기반의 인증시스템 (1)



(그림 12) SAML assertion 기반의 인증시스템 (2)

- VPN server가 사용자로부터 전송받은 assertion을 토대로 인증을 거쳐 사용자에게 인증 성공 여부를 브라우저를 통해서 알려준다. 이와 같이 사용자는 VPN 관리시스템 내 login server를 거쳐 인증하는 절차 없이 단 한번 로그인함으로써 다른 망의 VPN server에 인증이 가능하다.

3.2.3 VPN에서 artifact를 이용한 Single Sign On 과정

SAML artifact를 사용해서 single sign on을 제공하는 인증시스템의 인증 과정은 assertion이 아닌 artifact의 형태로 정보가 교환된다는 점이 이전의 인증 절차와 다른 점이다.

일단 VPN client(여기서는 사용자에 해당한다.)와 VPN server간 로그인이 성공적으로 이루어지면 client와 server간에는 동일한 세션(session)을 가진다. 동일하게 생성된 세션을 토대로 server는 이미 로그인을 한 client가 자신에게 접속을 요청할 때 client가 이전의 server에서 인증을 받았으며 자신의 server에 두 번째로 인증을 요구한다는 것을 알고 있다. 또한 artifact 전송 방식에서 두 번째 VPN server가 사용자가 선택한 server로 artifact를 전송받기 위해 redirection 하는 부분에서도, 이전 VPN server와 사용자 그리고 두 번째 server간 동일한 세션을 가지므로 이전 server는 artifact를 자동으로 URL에 실어 두 번째 server로 전송할 수 있게 된다.

4. 구현 및 성능분석

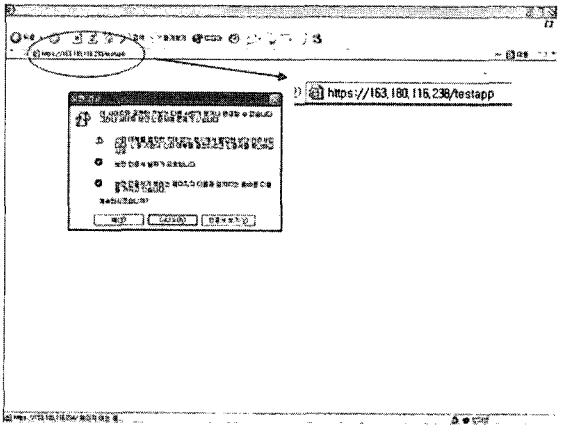
4.1 환경설정 및 구현

본 연구에서 제안된 SAML 기반의VPN server/client 간 인증 시스템을 다음과 같은 컴퓨팅 환경에서 설치과정을 거쳐 테스트하였다.

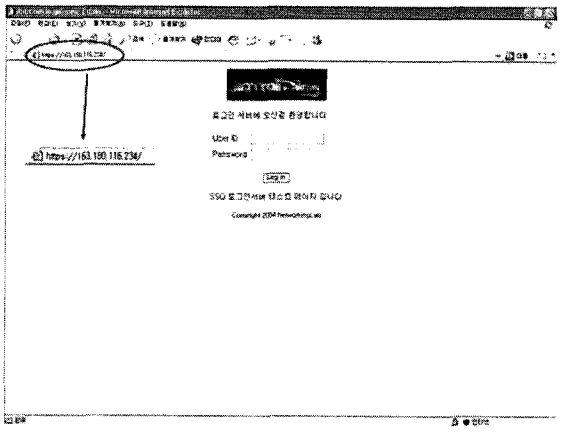
- Computer environment
  Compaq server 1.7GHz 256MB Redhat 9.0, Hancom Linux
- Language : C
- Web Server : Apache-1.3.32

처음 사용자가 특정 OSS의 정보를 얻고자 접속하고 URL은 https://163.180.116.238/testapp 이다. 인증이 필요하므로 OSS에 접근하기 이전에 VPN server에 접근한다.

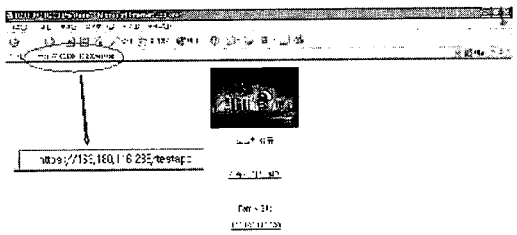
VPN server는 세션을 보고 사용자가 이전에 로그인했던 기록이 없으므로 login server에 로그인을 요청한다. login server는 로그인 입력 browser를 사용자에게 전송한다. 그림 13과 14는 그 과정을 나타낸다. 그림 13에서 접속을 하면 그림 14에서



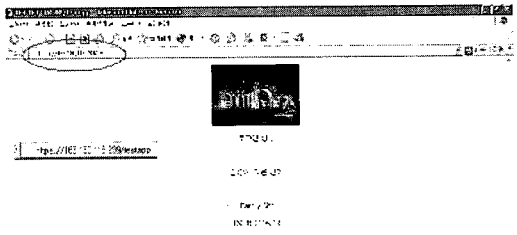
(그림 13) VPN server 접속 browser



(그림 14) VPN 관리시스템에서의 로그인 요청 Browser



(그림 15) 인증 성공 Browser



(그림 16) Single sign on을 통한 인증 성공 Browser

보는 바와 같이 https://163.180.116.234로 redirection되는 것을 볼 수 있다.

VPN server와 VPN 관리시스템 내의 login server간 관계가 유효함이 판명되면 login server는 사용자에게 로그인 요청을 하고 사용자는 아이디와 패스워드 정보를 전송한다. 이때 사용자가 보는 URL https://163.180.116.234는 VPN server의 browser가 아니라 VPN 관리시스템 내 login server의 browser이다. 인증을 거쳐 로그인에 성공하면 login server에서 assertion을 생성하고 VPN server로 전달한다. assertion을 전송받은 후 VPN server는 사용자에게 인증 성공 메시지를 보낸다. 사용자가 받은 인증 성공 browser는 그림 15와 같다. 그림 15에서 https://163.180.116.238/testapp는 login server에서 VPN server로 redirect 되었음을 보여준다.

후에 사용자가 다른 OSS에 접근하고자 또 다른 VPN server에 접속한다. 그림 15에서 browser 상에 다른 VPN server의 URL이 링크되어 있고 사용자는 원하는 사이트로 이동이 가능하다. 예를 들어 원하는 사이트의 URL이 https://163.180.116.239라고 한다. VPN server는 사용자 세션을 검사하여 이전에 이미 로그인했었던 사용자임을 알아내고 VPN 관리시스템 내 login server와 공유하는 키(key)를 사용해 인증 과정을 거친다. 그림 16는 인증 과정을 통해 로그인이 성공적으로 이루어졌음을 보여준다. 인증 성공 후 VPN server가 https://163.180.116.239로 redirect 하였음을 확인할 수 있다.

4.2 성능분석

성능을 분석하기 위해 설정한 환경은 다음과 같다.

- OPNET 10.5
- Server
  - HTTP Server
  - Received buffer : 8760 bytes
  - Max ACK delay : 0.2 seconds
  - Queuing : multi-server queue

사용자가 처음 로그인 후 세션과 assertion을 가지고 다른 OSS에 접속한 후 최종적으로 로그인 허가를 받기까지를 인증시간이라고 정의했을 때 SAML 기반이 아닐 때의 single sign on 인증시간과 SAML 기반으로 assertion, artifact를 사용하여 single sign on이 이루어지는 인증시간을 각각 측정하여 그 결과를 표 1에서 비교해보았다.

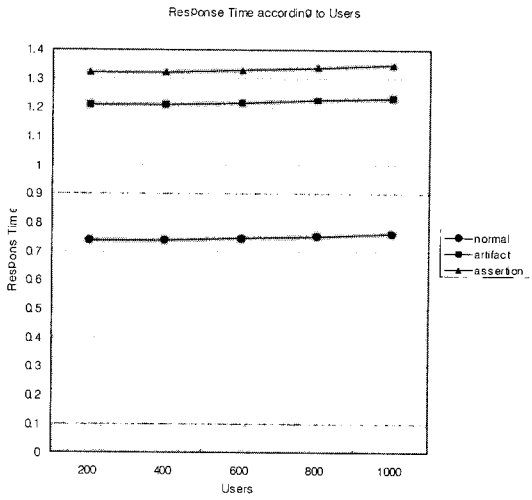
한 VPN server에 접속하는 동시 접속자 수를 200명, 400명, 600명, 800, 1,000명으로 나누고 각각 VPN server에 접속할 때 3가지 인증방식에 따라 측정한 동시 접속자 중 1인의 인증시간을 표 1로 나타내었다. 표 1에서 SAML 기반의 SSO 인증시간은 소수단위이지만 Non-SAML 기반의 SSO 인증시간보다 오래 걸리는 것을 볼 수 있다.

이는 일반 XML 데이터를 SAML로 변환하고 파싱하는 과정 때문이다. SAML 기반 SSO 인증시간을 측정한 결과 중에서도 SAML artifact 기반의

(표 1) 인증방법에 따른 SSO 인증시간 비교(second)

인증방식 접속자수(명)	Non-SAML	Artifact	Assertion
200	0.737412	1.209412	1.322412
400	0.737951	1.209951	1.322951
600	0.744989	1.216989	1.329989
800	0.753007	1.225007	1.338007
1,000	0.761819	1.233819	1.346819

인증시간이 assertion을 주고받으며 인증을 수행한 시간보다 짧은 인증시간이 요구되는 것을 볼 수 있다. 이것은 assertion은 XML기반의 텍스트 문서의 일종으로 포인터 개념인 artifact보다 데이터의 양이 많기 때문이다. 각 인증 방식 별 SSO 인증시간을비교한 그래프가 그림 17에 나타나 있어 그 차이를 비교해볼 수 있다.



(그림 17) 인증방식에 따른 SSO 인증시간 비교 그래프

5. 결 론

인터넷에서 정보 전달에 있어서 데이터자체의 안정성 및 사용자의 정보보호의 문제가 중요해짐에 따라 정보보호 기술이 주요 관심사로 떠오르게 되었다. 최근 전자문서들은 차츰 XML 정보보호기술 기반으로 표준화되어 전자상거래가 이루어지고 있다. XML 기반의 SAML은 비즈니스 거래 파트너들이 인증 정보, 권한 부여 정보, 프로파일 정보를 안전하게 교환할 수 있도록 설계된 것으로 새로운 보안 언어를 만든 것이 아닌 XML 기반으로 만들어져 XML에서 제공하는 장점들을 사용할 수 있다. XML디지털 서명을 이용해 메시지가 중간에 변경되는 일이 없고, 전송한 쪽의 정체를 명확하게 하고 있어 인증 정보 전달시 보안을 한층 강화시켰다. 본 논문에서는 기존의 인증 시스템에 대한 연구 동향을 분석하고 자료를 바탕으로 위의 장점을 가진 SAML을 실제 시스템에 적용시켜 VPN server들 간 Single Sign On을 제공할 수 있는 모델을 제시함으로써 사용자와 관리자 간 안전하게 인증 정보를 주고받을 수 있는 인증 시스템을 설계하였고 이를 구현하기 위한 방법론을 제시하였다. 이를 테스트하기 위해서 Linux 기반의 Compaq 서버에서 각 login server와 application server를 설치하였다. SAML을 기반으로 assertion과 artifact를 사용한 Single Sign On 인증시스템을 각각 구현하고 성능을 비교 분석하였다.

SAML assertion 구조는 공백을 허용하지 않기 때문에 공격자가 이를 악용하여 공백 문자를 무차별적으로 보내면 해당 시스템에서 에러로 인지하고 에러 메시지를 계속 보낸다. 그러므로 트래픽 부하가 걸릴 위험이 있다. 향후에 SAML assertion을 생성하여 데이터를 전송할 때 위와 같은 SAML의 취약점으로 인해 일어날 수 있는 문제점을 해결할 수 있도록 연구가 필요하다.

참 고 문 헌

[1] Berket, K., Essiari, A., Muratas, A.: PKI-based security for peer-to-peer information sharing, Peer-to-Peer Computing, 2004. Proceedings. Fourth International Conference on, Pages:45 52, Aug. 2004

[2] Gross, T, "Security analysis of the SAML

single sign-on browser/artifact profile", Computer Security Applications Conference 2003, Proceedings. 19th Annual, Pages:298 - 307, 2003

[3] Gary Ellison, Jeff Hodges, Susan Landau, "Security and Privacy Concerns of Internet Single Sign-On", Liberty v1.6, September 2002

[4] Jan De Clercq, "Single Sign-On Architectures", Proceedings of the International Conference on Infrastructure Security table of contents, Pages:40-58, 2002

[5] Michael Fleming Grubb and Rob Carter: Single Sign-On and the System Administrator, Proceedings of the Twelfth Systems Administration Conference(LISA'98), 1998

[6] Nayef Abu-Ghazaleh, Michael J. Lewis, "Differential Serialization for Optimized SOAP Performance", In proceedings of 13th International Symposium on High Performance Distributed Computing (HPDC), Honolulu, Hawaii, pp: 55-64, June 2004

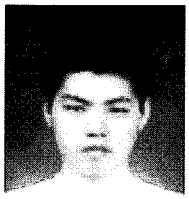
[7] Miyoshi, J.; Ishii, H., "Network-based single sign-on architecture for IP-VPN", Communications, Computers and signal Processing, 2003. PACRIM. 2003 IEEE Pacific Rim Conference on, Pages:458-461 vol.1, Volume:1, 28-30 Aug. 2003

[8] Takeda, T., Kojima, H, Inoue, I., "Optical VPN architecture and mechanisms", Communications, 2003. APCC 2003. The 9th Asia-Pacific Conference on, Pages:751 - 755 Vol.2, Volume:2, 21-24 Sept. 2003

[9] Qiu Xuesong; Xiong Ao; Meng Luoming, "The study and implementation the VPN service management system", Computers and Communications, 2000. Proceedings. ISCC 2000. Fifth IEEE Symposium on, Pages:66-71, 3-6 July 2000

[10] Cohen, R., "On the establishment of an access VPN in broadband access networks", Communications Magazine, IEEE, Pages: 156-163, Volume:41, Issue:2, Feb. 2003

◐ 저 자 소 개 ◑



강 명 수

2004년 2월 경희대학교 컴퓨터공학과 학사
2006년 2월 경희대학교 컴퓨터공학과 석사 졸업예정
관심분야 : 네트워크 보안
E-mail : mskang@networking.khu.ac.kr

홍 충 선

1983년 2월 경희대학교 전자공학과 졸업
1985년 8월 경희대학교 전자공학과 석사
1988년 3월~1999년 8월 한국통신망 연구소 선임연구원 / 네트워킹 연구실장
1997년 3월 Dept. of Information and Computer Science, Keio University (공학박사)
1999년 9월~현재 경희대학교 전자정보학부 부교수
관심분야 : 차세대 인터넷, 센서네트워크, 네트워크 보안, 네트워크 QoS
E-mail : cshong@khu.ac.kr