논문 17-42-04-18 The Journal of Korean Institute of Communications and Information Sciences '17-04 Vol.42 No.04
https://doi.org/10.7840/kics.2017.42.4.838

CNG 암호 라이브러리의 보안 취약점 분석

이 경 률◆, 오 인 수*, 이 선 영*, 임 강 빈°

Vulnerability Analysis on the CNG Crypto Library

Kyungroul Lee◆, Insu Oh*, Sun-Young Lee*, Kangbin Yim°

요 약

최근 마이크로소프트사의 CAPI를 대체하기 위해 제안된 CNG는 플러그인 구조 기반의 독립된 모듈들로 구성되어 있기 때문에 개발비용과 확장 용이성 부분에서 우수하다. 하지만 이러한 이점과 반대로 보안성에 대한 고려는 다소 부족하며, 현재 CNG가 배포되어 활용되고 있는 상황에서 이와 관련된 연구는 반드시 필요하다. 이에 본 논문에서는 CNG에서 발생 가능한 보안 취약점을 분석하였다. 분석된 취약점을 토대로 개념 검증 도구를 구현하여 이를 검증하였다. 검증 결과는 CNG를 활용하는 Outlook 프로그램과 Internet Explorer 프로그램에서 메일 및 계정정보의 탈취, Amazon, E-bay, Google, Facebook 웹 사이트의 계정정보의 탈취가 가능하였다. 본 논문의 결과는 CNG를 활용하는 다양한 응용의 보안성을 향상시키는데 기여할 것으로 사료된다.

Key Words : CNG library, Crypto library, Crypto API, Vulnerability, Analysis

ABSTRACT

CNG which was released as a substitute of the previous CAPI (Cryptography API) library from Microsoft is constructed with individual modules based on the plug-in architecture, this means CNG is exceedingly helpful in the cost of development as well as the facility of extension. On the opposite side of these advantages, considerations on security issues are quite insufficient. Therefore, a research on security assurance is strongly required in the environment of distributing and utilizing the CNG library, hence, we analyze possible security vulnerabilities on the CNG library. Based on analyzed vulnerabilities, proof-of-concept tools are implemented and vulnerabilities are verified using them. Verified results are that contents of mail, account information of mail server, and authentication information of web-sites such as Amazon, E-bay, Google, and Facebook are exposed in Outlook program and Internet Explorer program using CNG library. We consider that the analyzed result in this paper can improve the security for various applications using CNG library.

Ⅰ. 서 론

최근 다양한 보안 응용 프로그램에서 구현상의 취약점으로 인한 해킹사고가 잇따르고 있다. 이러한 해킹사고는 사소하게는 개인정보의 유출부터 심각하게는 금전적 피해를 동반하는 온라인 사기, 국가안보를 위협할만한 기밀시설의 공격 등에 이르기까지 그 치명성이나 규모가 다양하며, 이러한 사이버 위협은 사회 각 구성원 누구나 목표물이 될 수 있다는 점에서 모두가 관심을 가져야 할 사안이다.

※ 본 연구는 한국연구재단 이공분야기초연구사업(No. NRF-2015R1D1A1A01057300) 지원 및 순천향대학교 산학협력단 관리로 수행되었습니다.

◆ First Author : Soonchunhyang University R&BD Center for Security and Safety Industries (SSI), carpedm@sch.ac.kr, 정회원
° Corresponding Author : Soonchunhyang University Dept. of Information Security Engineering, yim@schac.kr, 정회원
* Soonchunhyang University Department of Information Security Engineering, {catalyst32, sunlee}@sch.ac.kr
논문번호 : KICS2017-02-056, Received February 27, 2017; Revised March 27, 2017; Accepted March 29, 2017

보안 응용 프로그램에서 암호의 중요성은 아무리 강조하여도 지나치지 않다. 그러나 고도의 보안성을 지원하기 위하여 필요한 기반기술도 그 구현상에서 구조적 또는 기능적 취약점이 포함되고 있을 수 있으며, 이러한 경우, 첨단 보안 응용 프로그램도 사이버 위협에 대하여 무용지물이 될 수 있다[9-11]. 따라서 보안구조의 가장 기본적인 요소인 암호 라이브러리의 보안 취약점 분석 및 공격 가능성 진단은 보안 응용 프로그램의 개발 및 구현과정에서 가장 원천적이고 중요한 설계요소라고 볼 수 있다.

CNG(Cryptography API: Next Generation)는 윈도우 비스타부터 등장하였으며, 윈도우 비스타, 7, 8, 서버 2008, 서버 7, 스토리지 서버 2008 등에 기본으로 제공한다[1]. 특히, 윈도우 8부터는 CNG가 핵심적인 요소 중 하나로 제공되고 있으며, 윈도우 비스타 출시 이후로 윈도우 XP의 점유율이 줄어들고 윈도우 비스타와 7 이후 버전의 점유율이 증가되고 있다. 또한, 마이크로소프트 오피스 2010에서 디지털 서명을 사용한 정보보안 및 무결성을 향상시키기 위한 방법으로 CNG를 사용하고 있으며 [2], 윈도우 8부터 클라우드 컴퓨팅과 관련된 일환으로 데이터 보호를 위하여 CNG의 DPAPI(Data Protection Application Programming Interface)가 새로이 등장하여 암호 시스템의 핵심 기술로 소개되고 있다[3]. 이는 CNG가 활용되는 환경이 증가하는 것을 의미하므로 이에 대한 연구가 시급한 실정이다[4-7].

Ⅱ. DLL 인젝션을 이용한 함수 후킹

DLL 인젝션을 이용한 함수 후킹은 우선 CNG를 활용하는 샘플 프로그램을 작성하여 그 프로그램에 대한 공격 가능성을 확인하고, 공격 가능성이 확인되면 윈도우즈 기반의 시스템에서 CNG를 활용하는 응용 프로그램을 조사한 후, 샘플 프로그램에 시도한 공격을 조사한 응용 프로그램에 적용하여 취약점을 도출하는 것으로 진행한다. 실험환경은 Intel(R) Core(TM) i5-2410M CPU 2.30 GHz, 4GB RAM, Microsoft Windows 7 Home Premium K 운영체제가 설치된 PC를 이용하였으며, CNG를 사용하기 위하여 CNG SDK, Windows SDK 7.1과 .NET Framework 4를 설치하였으며, 개념검증도구를 구현하기 위하여 Visual Studio 2005를 설치하고, CNG의 샘플 소스 참조를 위하여 CNG Demo Sample Code를 설치하였다.

2.1 샘플 프로그램 작성

CNG에서의 후킹 가능성을 확인하기 위하여 CNG를 활용하는 샘플 프로그램을 작성하였다. 우선 윈도우즈 7에서 CNG가 사용하는 함수를 조사하였으며, CNG가 BCrypt와 NCrypt 라이브러리로 분류되었으므로 각 라이브러리에서 활용 가능한 함수 리스트를 그림 2와 3에 나타내었다[8]. BCrypt 함수는 BCryptEncrypt, BCryptDecrypt 등의 49가지 함수를 지원하며, NCrypt 함수는 SslEncryptPacket, SslDecryptPacket 등의 54가지 함수를 지원한다.

샘플 프로그램은 응용 프로그램과 마찬가지로 CNG가 지원하는 함수를 활용하기 위하여 BCrypt와 NCrypt 라이브러리를 로드한 후, 테스트 요청이 있을 경우 요청을 원하는 함수를 호출하도록 구성하였다. 응용 프로그램에서 암/복호 혹은 기타 기능의 수행을 샘플 프로그램에서는 버튼을 클릭하는 것으로 가정하였고, 그 기능의 수행은 호출한 함수명을 출력하는 것으로 가정하였다. CNG를 활용하는 샘플 프로그램 UI를 그림 1, BCrypt 함수 호출 결과를 그림 2, NCrypt 함수 호출 결과를 그림 3에 나타내었다.

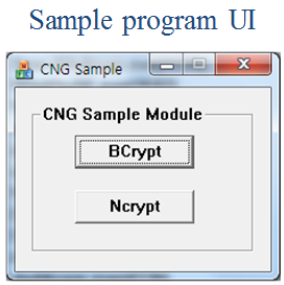

그림 1. 샘플 프로그램 UI
Fig. 1. Sample program UI

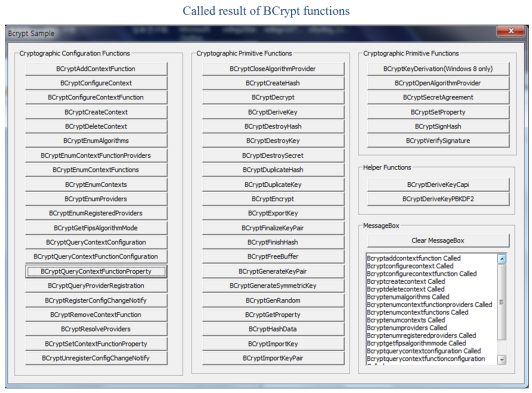

그림 2. BCrypt 함수 호출 결과
Fig. 2. Called result of BCrypt functions

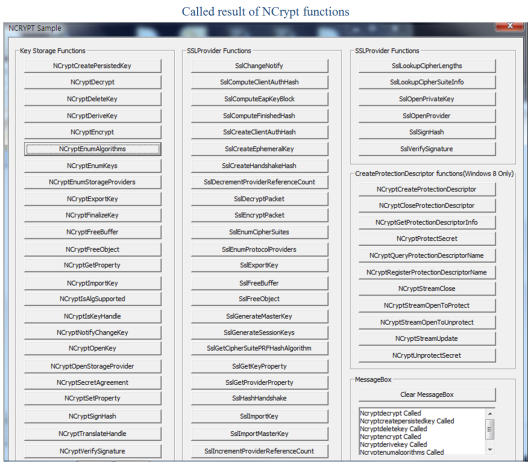

그림 3. NCrypt 함수 호출 결과
Fig. 3. Called result of NCrypt functions

2.2 공격 프로그램 작성

취약점 확인을 위한 공격 대상 프로그램인 샘플 프로그램을 구현하였으므로 구현된 샘플 프로그램을 공격하기 위한 공격 프로그램을 작성하였다. 일반적으로 공격자가 특정 프로그램을 공격하기 위해서는 공격 대상 프로그램에 대한 메커니즘 및 동작과정에 대한 분석이 선행되어야 하지만, 이 과정은 공격 이전에 공격 가능성에 대한 확인이므로 샘플 프로그램에 대한 분석이 이미 완료되었고, 공격 프로그램은 사용자 PC에 설치되었다고 가정하였다.

공격 프로그램의 동작과정은 다음과 같다. 샘플 프로그램에 후킹을 위한 악의적인 DLL을 인젝션하면, 샘플 프로그램에 악의적인 DLL이 로드되고 악의적인 DLL은 BCrypt와 NCrypt 라이브러리 내의 모든 함수를 후킹한다. 후킹이 완료된 후, 샘플 프로그램에서 BCrypt와 NCrypt 라이브러리 내의 특정 함수를 호출하도록 요청할 경우, 공격이 성공적이라면 사용자가 요청한 BCrypt와 NCrypt 라이브러리 내의 특정 함수를 후킹한 공격 프로그램의 함수가 호출되고, 실패한다면 사용자가 요청한 BCrypt와 NCrypt 라이브러리 내의 원래 함수가 호출된다. 따라서 공격에 성공한다면, 후킹한 함수는 모든 악의적인 기능을 수행할 수 있으므로 취약점이 존재한다고 할 수 있어 공격에 대한 가능성을 확인할 수 있다. 실험을 위하여 악의적인 기능을 하는 후킹 함수는 실제로 악의적인 기능은 수행하지 않고 해당 함수가 후킹되었다는 메시지를 출

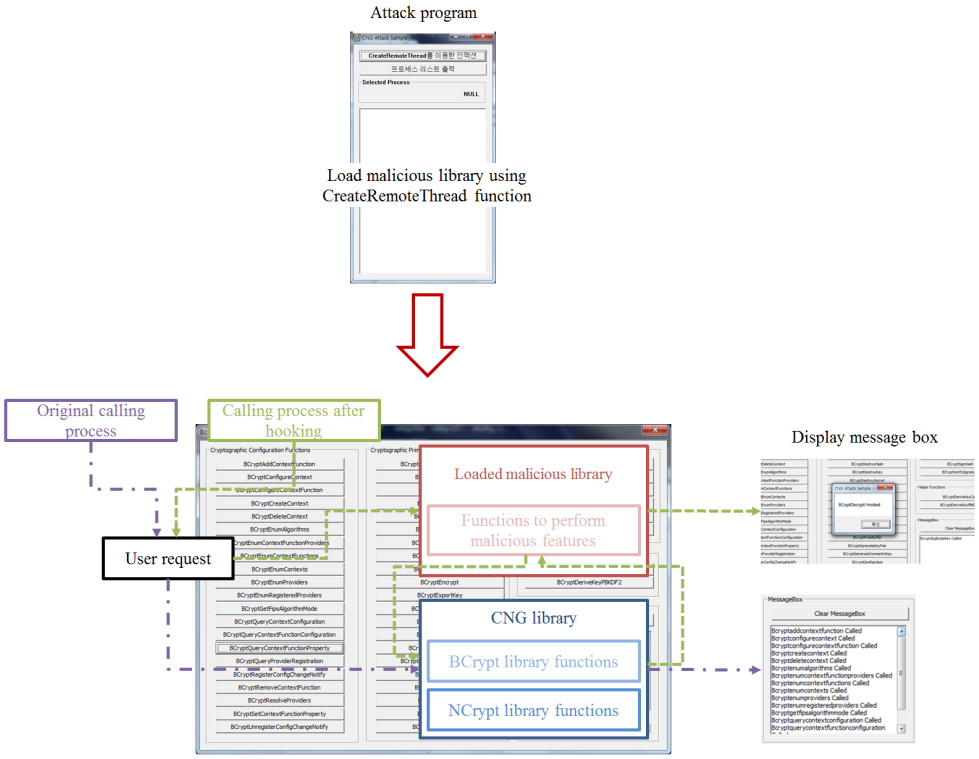

그림 4. DLL 인젝션을 이용한 CNG 후킹 시 동작과정
Fig. 4. The operation process by DLL injection

력하는 것으로 대체하였다. 다시 말하면, 공격에 성공할 경우, 후킹되었다는 메시지를 출력할 것이고, 공격에 실패할 경우, 요청한 BCrypt와 NCrypt 라이브러리 내의 특정 함수명을 출력할 것이다. 공격 프로그램의 동작과정을 그림 4에 나타내었다.

후킹을 위해서는 시스템에서 동작 중인 후킹 대상 프로세스를 지정하거나 전체 프로세스에 후킹을 시도할 수 있지만, 본 논문에서는 대상 프로세스를 선택하는 방법으로 구현하였으며, 프로세스 선택을 위하여 "프로세스 리스트 출력" 버튼을 클릭하면, 시스템에서 동작 중인 모든 프로세스를 출력하고, 공격자가 대상 프로세스를 선택한 후, "CreateRemoteThread를 이용한 인젝션" 버튼을 클릭함으로써 선택한 대상 프로세스에 악의적인 DLL이 로드되면서 후킹을 시도한다. 이러한 동작을 하는 공격 프로그램 UI를 그림 5, 공격

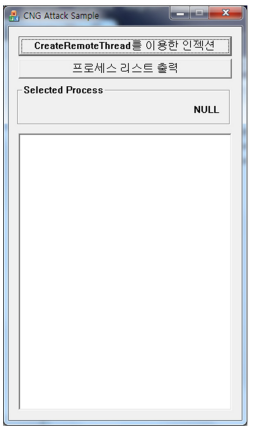

그림 5. 공격 프로그램 UI
Fig. 5. Attack program UI

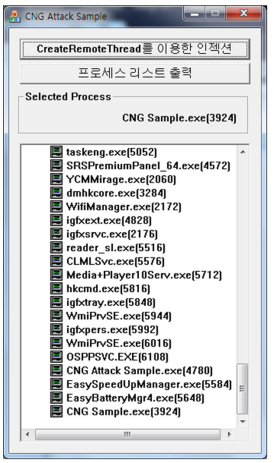

그림 6. 후킹 대상 프로세스 선택
Fig. 6. Selection of the process for hooking

을 시작하기 위하여 후킹 대상 프로세스를 선택하는 화면을 그림 6, 공격 결과를 그림 7에 나타내었다. DLL 인젝션을 이용한 CNG의 후킹 공격 실험 결과, 후킹이 성공적으로 수행되어 메시지 박스를 출력하는 것을 확인하였다.

2.3 응용 프로그램의 공격 가능성 확인

상기와 같은 실험 결과를 통하여 본 논문에서는 후킹을 이용한 CNG 라이브러리의 공격 가능성을 확인하였고, 샘플 프로그램이 아닌 실제 시스템에서 구동중인 응용 프로그램에서도 동일하게 공격을 시도하였다. 우선 윈도우즈 7 32비트 운영체제가 설치된 시스템에서 BCrypt와 NCrypt 라이브러리를 활용하는 응용 프로그램을 조사한 결과, Outlook 프로그램과 Internet Explorer 프로그램이 CNG를 활용하는 것을 확인하였으며, 그 결과를 그림 8에 나타내었다.

일반적으로 공격을 위해서는 동작과정 및 메커니즘의 분석이 선행되어야 하고, 분석 결과를 토대로 공격을 시도하는 단계로 이루어진다. 하지만 샘플 프로그램의 경우에는 공격의 가능성을 확인하기 위한 테스트이므로 분석이 따로 요구되지 않아 수행하지 않았지만, 응용 프로그램의 경우에는 이와 같은 분석 결과가 필요하므로 DLL 인젝션을 이용하여 후킹을 시도한 후, 그 결과를 분석함으로써 공격 가능성을 확인하였다.

Ⅲ. Outlook 프로그램의 공격 가능성 확인

우선, Outlook 프로그램을 대상으로 동작과정 및 메커니즘의 분석을 시도하였으며, Outlook 프로그램에 DLL 인젝션을 시도한 후, BCrypt와 NCrypt 라이브러리에서 활용하는 모든 함수를 후킹하고 메일을 보낼 경우의 입/출력 인자를 확인하였다. 그 결과 SslEncryptPacket 함수에서 메일과 관련된 정보를 암호화하는 것을 확인하였고, 이를 그림 9에 나타내었다[12].

입력 인자를 확인한 결과, pbInpt 인자에 메일과 관련된 정보를 MIME 형태로 전송하는 것을 확인하였으며, 보내는 사람, 받는 사람, 제목, 보낸 날짜, 참조, 메일 내용 등을 포함하였다. 이는 Outlook 프로그램이 메일을 전송할 때, SSL 통신을 이용하여 메일과 관련된 정보를 암호화하여 메일 서버로 전송하는 것이라 판단되므로, 공격자는 메일 정보를 암호화하는 SslEncryptPacket 함수를 후킹하여 메일과 관련된 정보를 탈취할 수 있으며, 탈취한 정보를 공격자의 서버로 전송함으로써 자동화된 공격을 시도할 수 있을 것

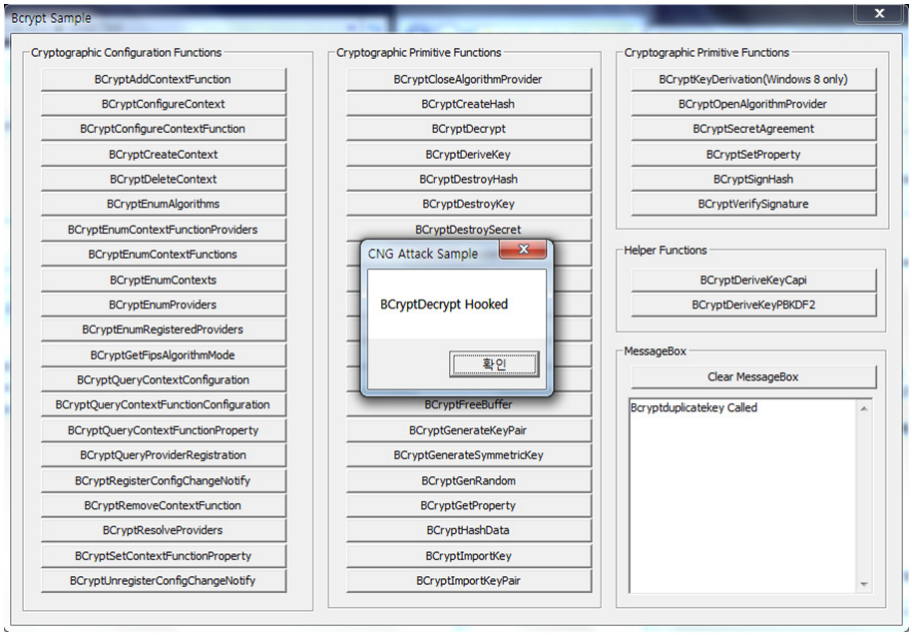

그림 7. DLL 인젝션을 이용한 후킹 공격 결과
Fig. 7. Hooking result using DLL injection

으로 판단된다. 따라서 본 논문에서는 상기 공격 시나리오를 토대로 Outlook 프로그램이 메일을 전송할 때, 보내는 사람, 받는 사람, 참조, 제목, 날짜, 내용을 탈취하여 공격자에게 전송하고, 공격자가 이를 수신하여 탈취한 정보를 확인하는 공격 시나리오를 구성하였으며, Outlook 프로그램의 공격 프로그램과 보내는 메일의 내용을 그림 10, Outlook 프로그램에서 보내는 메일을 탈취한 결과를 그림 11에 나타내었다.

후킹을 이용하여 Outlook 프로그램에서 보내는 메일에 대한 공격 가능성을 확인하였으므로, 동일한 공격을 시도하여 받는 메일의 탈취 가능성을 확인하였다. 하지만 받는 메일의 경우, 그림 12와 같이

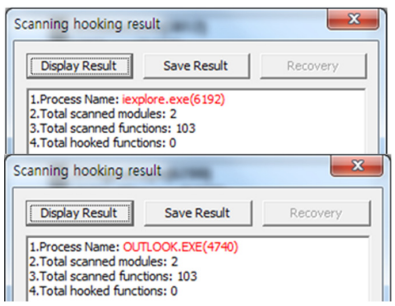

그림 8. CNG를 활용하는 응용 프로그램 확인 결과
Fig. 8. Surveyed results of application programs using CNG

BCryptDecrypt의 출력 인자에 메일에 대한 리스트만 출력되고 메일의 내용은 출력되지 않았다. 따라서 메일 정보 이외에 다른 정보를 활용할 것이라 판단하여 입/출력 정보를 확인한 결과, Outlook 프로그램은 설정된 메일 서버로부터 메일을 수신하기 위하여 메일 서버에 접속하여야 하며, 메일 서버에 접속하기 위하

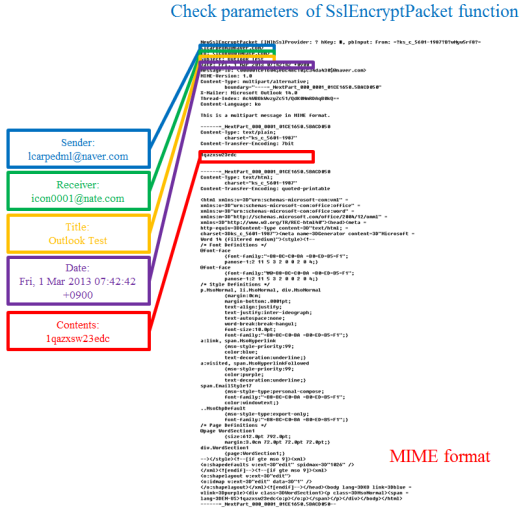

그림 9. Outlook 프로그램에서 메일을 보낼 경우의 메일과 관련된 인자
Fig. 9. Mail-related parameters when sending on Outlook program

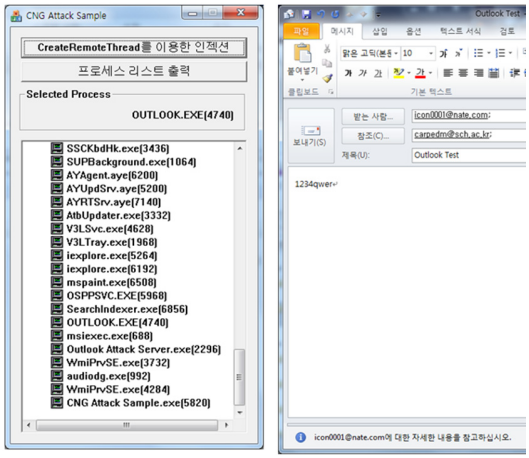

그림 10. Outlook 프로그램의 메일 탈취 프로그램 UI와 메일 전송 내용
Fig. 10. Mail attack program UI and sending contents of Outlook program

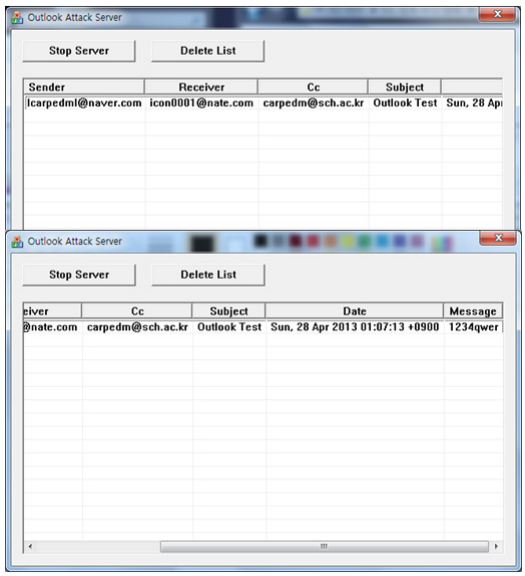

그림 11. Outlook 프로그램에서 보내는 메일 탈취 결과
Fig. 11. Stolen result of sending mail of Outllook program

여 그림 13과 같이 계정정보를 전송하는 것을 확인하였다. 이 결과를 토대로 Outlook 프로그램이 메일 서버로부터 메일을 수신할 때, 전송하는 계정정보를 탈취할 수 있을 것으로 판단하였으며, 탈취한 계정정보를 공격자에게 전송하여 공격자가 확인하는 것으로 공격 가능성을 검증하였다.

실험을 위하여 보내는 메일을 탈취하는 공격과 마찬가지로 메일의 계정정보를 탈취하여 공격자에게 전송하는 DLL을 Outlook 프로그램에 인젝션하기 위한

```
NewBCryptDecrypt [OUT]1 31663
2 27700
3 23069
4 25174
5 25587
6 25648
7 9210
8 26085
9 24062
10 116328
11 1761
12 1934
13 2284
14 1996
15 2050
16 26150
17 2110
18 2381
19 24562
20 25485
21 24562
22 84560
23 22966
24 1646
25 10999
```

그림 12. Outlook 프로그램이 메일을 수신할 때 노출되는 메일 리스트
Fig. 12. Exposed mail list when Outlook program receives mail

```
NewSslEncryptPacket [IN]USER lcarpedml
NewSslEncryptPacket [IN]PASS lisa.sch.ac.kr
NewSslDecryptPacket [OUT]+OK NMQpopper NaverMail Team 2004 base Qpopper
(version naver-RB-13.03.14@31356(libnm:20121108@24693)) at tpop04-
2.nm.nhnsystem.com starting.
```

그림 13. Outlook 프로그램이 메일을 수신할 때 전송하는 계정정보
Fig. 13. Transferred account information when Outlook program receives mail

공격 프로그램과 탈취한 계정정보를 출력하는 공격자의 서버로 구성하였으며, 공격 프로그램의 UI와 Outlook 프로그램에서 탈취한 메일의 계정정보를 그림 14에 나타내었다.

Ⅳ. Internet Explorer 프로그램의 공격 가능성 확인

Outlook 프로그램에서의 공격과 마찬가지로 공격 가능성을 확인하기 위하여 Internet Explorer 프로그램을 대상으로 동작과정 및 메커니즘의 분석을 시도하였으며, Internet Explorer 프로그램에 DLL 인젝션을 시도한 후, BCrypt와 NCrypt 라이브러리에서 활용하는 모든 함수를 후킹하고 SSL 통신으로 웹 사이트에 접속할 경우의 입/출력 인자를 확인하였다. 그 결과, SslEncryptPacket 함수에서 사용자가 접속하는 웹 사이트의 계정정보가 암호화되는 것을 확인하였고, 이를 그림 15에 나타내었다[12].

입력 인자를 확인한 결과, pbInput 인자에 사용자의 계정정보를 사이트마다 다르게 전송하는 것을 확인하였으며, Amazon의 경우 email 필드에 아이디, password 필드에 비밀번호, E-bay의 경우 userid 필드

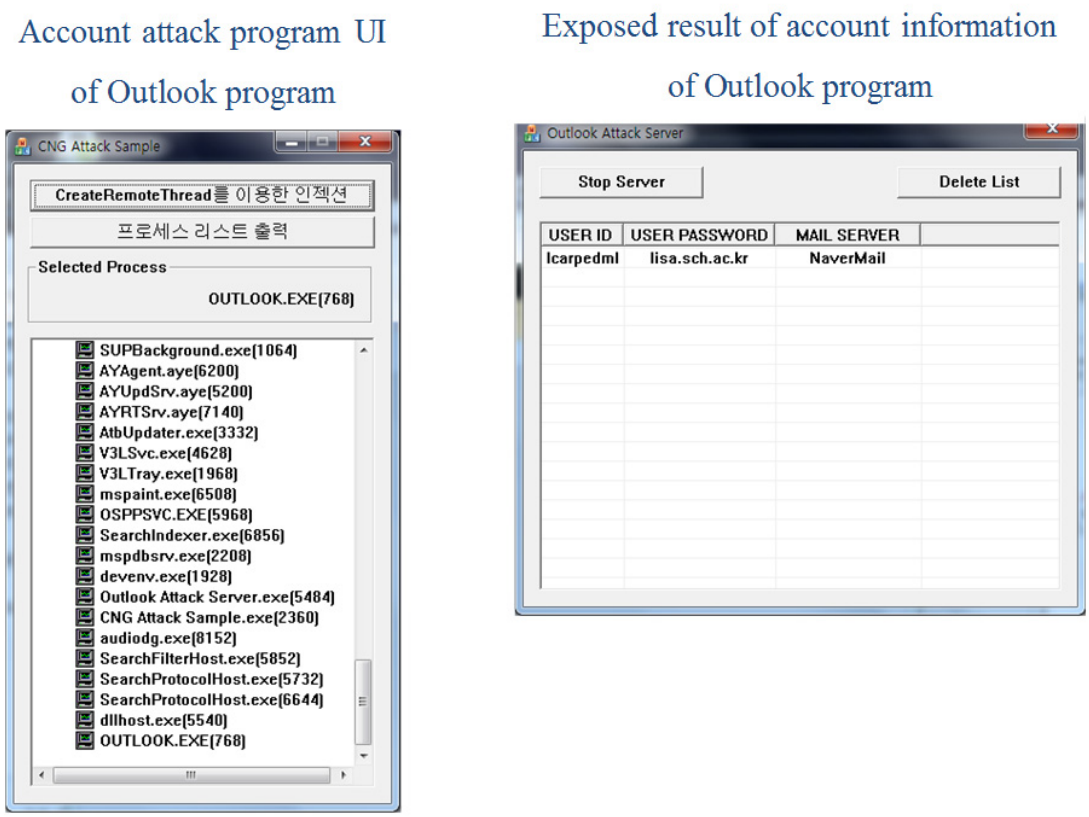

그림 14. Outlook 프로그램의 계정 탈취 프로그램 UI와 계정 탈취 결과
Fig. 14. Account attack program UI and exposed result of account information of Outlook program

에 아이디, pass 필드에 비밀번호, Google의 경우 Email 필드에 아이디, Passwd 필드에 비밀번호, Facebook의 경우 email 필드에 아이디, pass 필드에 비밀번호가 포함되었다. 이는 Internet Explorer 프로그램이 웹 사이트에 접속할 때, SSL 통신을 이용하여 사용자 계정정보를 암호화함으로써 사용자 계정정보를 보호하는 것이라 판단된다. 따라서 공격자의 경우, 계정정보를 암호화하는 SslEncryptPacket 함수를 후킹하여 계정과 관련된 정보를 탈취할 수 있으며, 탈취한 정보를 공격자의 서버로 전송함으로써 자동화된 공격을 시도할 수 있다. 이에 본 논문에서는 상기 공격 시나리오를 토대로 사용자가 Internet Explorer 프로그램을 이용하여 Amazon, E-bay, Google, Facebook 사이트에 접속할 때, 사용자의 아이디와 비밀번호를 탈취하여 공격자에게 전송하고, 공격자가 이를 수신하여 탈취한 정보를 확인하는 것으로 공격 가능성을 검증하였으며, Internet Explorer 프로그램의 공격 프로그램과 탈취한 웹 사이트 계정정보를 그림 16에 나타내었다.

CNG에서 DLL 인젝션을 이용한 후킹 공격의 실험 결과는 Outlook 프로그램에서 메일 서버로 보내는 메일의 탈취가 가능하고, 메일 서버로부터 메일을 수신할 경우 메일 서버로 접속하는 계정정보의 탈취가 가능한 것을 검증하였다. 또한 Internet Explorer 프로그램에서 SSL 통신을 이용하여 Amazon, E-bay, Google, Facebook 웹 사이트에 접속할 경우, 사용자가 입력하는 계정정보의 탈취가 가능한 것을 검증하였다. 이와 같은 공격 도구를 안티 바이러스 제품이 적절하게 대응하는지 확인을 위하여 국내 안티 바이러스 제품인 알약과 V3를 기반으로 정적, 동적 검사를 실시한 결과, 두 검사 모두 바이러스로 탐지되지 않아 그 대응책이 시급한 것이라 판단된다. 안티 바이러스 제품은 사용자가 쉽게 설치할 수 있는 제품인 알약의 공개용 버전, V3의 lite 버전으로 검사하였으며, 검사 결과를 그림 17에 나타내었다.

그림 15. Internet Explorer 프로그램에서 웹 사이트 접속 시 노출되는 계정정보
Fig. 15. User account related parameter when login

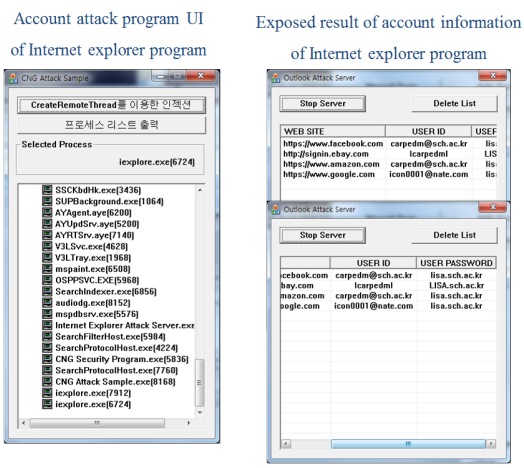

그림 16. Internet Explorer 프로그램의 계정 탈취 프로그램 UI와 계정 탈취 결과
Fig. 16. Account attack program UI and exposed result of account information of Internet explorer program

그림 17. DLL 인젝션을 이용한 후킹 도구의 바이러스 검사 결과
Fig. 17. Detection result of attack tools using DLL injection

V. 결 론

본 논문은 CNG에서의 보안 취약점에 대하여 연구하였다. 이를 위하여 CNG를 활용하는 공격 소프트웨어와 취약점 검증 소프트웨어를 구현하였고, 그 결과를 토대로 발생 가능한 취약점을 도출하였다. 대표적인 공격 기법인 후킹을 활용하여 응용 프로그램 중 Outlook 프로그램과 Internet Explorer 프로그램의 공격 가능성을 분석하고 개념 검증 도구를 구현하였다. 구현된 도구를 통하여 실험한 결과를 살펴보면, Outlook 프로그램에서 메일 서버로 보내는 메일의 탈취가 가능하고, 메일 서버로부터 메일을 수신할 경우, 메일 서버로 접속하는 계정정보의 탈취가 가능한 것을 검증하였다. 또한, Internet Explorer 프로그램에서 SSL 통신을 이용하여 Amazon, E-bay, Google, Facebook 웹 사이트에 접속할 경우, 사용자가 입력하는 계정정보의 탈취가 가능한 것을 검증하였다. 본 논문의 결과는 CNG를 활용하는 다양한 응용의 보안성을 향상시키는데 기여할 것으로 사료된다.

References

[1] Microsoft, *Cryptography Next Generation*, Retrieved Jan., 23, 2017, from http://technet.microsoft.com/en-us/library/cc730763(v=ws.10).aspx

[2] Microsoft, *Business Productivity at Its Best – Office 2010 and SharePoint 2010 white paper*, Retrieved Jan., 23, 2017, from http://technet.microsoft.com/en-us/library/ff384150(v=office.14).aspx

[3] Microsoft, *CNG DPAPI*, Retrieved Jan., 23, 2017, from http://msdn.microsoft.com/en-us/library/windows/desktop/hh706794(v=vs.85).aspx

[4] A. Young, "Cryptoviral extortion using Microsoft's Crypto API," *J. IJIS*, vol. 5, no. 2, pp. 67-76, Apr. 2006.

[5] Symantec, *How Trojan.Zbot.B!inf Uses Crypto API*, Retrieved Jan., 23, 2017, from http://www.symantec.com/connect/blogs/how-trojanzbotbinf-uses-crypto-api

[6] A. Young and M. Yung, *An implementation of cryptoviral extortion using microsoft's crypto api*, Retrieved Jan., 23, 2017, from http://www.cryptovirology.com/cryptovfiles/newbook/Chapter2.pdf

[7] J. Song and I. Hwang, "A study on neutralization malicious code using Windows Crypto API and an implementation of Crypto API hooking tool," *J. KIISC*, vol. 21, no. 2, pp. 111-117, Apr. 2011.

[8] K. Lee, Y. Lee, J. Park, I. You, and K. Yim, "Security Issues on the CNG Cryptography Library (Cryptography API: Next Generation)," in *Proc. IMIS*, pp. 709-713, Taichung, Taiwan, Jul. 2013.

[9] W.-N. Kim, M.-S. Jang, J. Seo, and S. Kim, "Vulnerability discovery method based on control protocol fuzzing for a railway SCADA system," *J. KICS*, vol. 39, no. 4, pp. 362-369, Apr. 2014.

[10] H. J. Kwon and S. J. Kim, "RFID distance bounding protocol secure against mafia and terroist fraud," *J. KICS*, vol 39, no. 11, pp. 660-674, Nov. 2014.

[11] Y.-H. Goo, S.-O. Choi, S.-K. Lee, S.-M. Kim, and M.-S. Kim, "Tracking the source of cascading cyber attack traffic using network traffic analysis," *J. KICS*, vol. 41, no. 12, pp. 1771-1779, Dec. 2016.

[12] K. Lee, I. You, and K. Yim, "Vulnerability analysis on the CNG crypto library," in *Proc. IMIS*, pp. 221-224, Blumenau, Brazil, Jul. 2015.

이 경 룰 (Kyungroul Lee)

2008년 8월 : 순천향대학교 정보보호학과(공학사)
2010년 8월 : 순천향대학교 정보보호학과(공학석사)
2015년 2월 : 순천향대학교 정보보호학과(공학박사)
2011년 5월~2011년 12월 : (미) 퍼듀대학교 방문연구원
2015년 6월~2016년 2월 : 순천향대학교 박사후연구원
2016년 3월~현재 : 순천향대학교 연구조교수
<관심분야> 취약점 분석, 시스템 보안, 하드웨어 보안, 인터넷 뱅킹, 사용자 인증, 디바이스 인증

오 인 수 (Insu Oh)

2012년 3월~현재 : 순천향대학교 정보보호학과 학사과정
<관심분야> 취약점 분석, 디바이스 분석, 사물인터넷 보안, 모바일 보안

이 선 영 (Sun-Young Lee)

1993년 2월 : 부경대학교 전자계산학과(이학사)
1995년 2월 : 부경대학교 전자계산학과(이학석사)
2001년 3월 : 일본동경대학 전자정보공학(공학박사)
2004년 3월~현재 : 순천향대학교 정보보호학과 교수
<관심분야> 콘텐츠 보안, 암호이론, 정보이론, 정보보안

임 강 빈 (Kangbin Yim)

1992년 2월 : 아주대학교 전자공학과(공학사)
1994년 2월 : 아주대학교 전자공학과(공학석사)
2001년 2월 : 아주대학교 전자공학과(공학박사)
1999년 3월~2000년 2월 : (미)아리조나주립대학교 연구원
2003년 3월~현재 : 순천향대학교 정보보호학과 교수
2005년 3월~현재 : 한국정보보호학회 이사
2009년 3월~현재 : 한국인터넷정보학회 이사
2010년 12월~2012년 2월 : (미)퍼듀대학교 객원교수
<관심분야> 취약점 분석, 내부자 공격, 보안 하드웨어 구조, 인증 프로토콜, 홈랜드 시큐리티