공공 Wi-Fi의 Evil Twin 공격 대응을 위한 RADIUS 기반 계정 생성 시스템 개발 및 구현

김명규[1], 전상훈[2*]
[1]수원대학교 컴퓨터SW학과 학생, [2]수원대학교 정보보호학과 교수

Development and Implementation of a RADIUS-Based Account Generation System for Mitigating Evil Twin Attacks on Public Wi-Fi

Myeonggyu Kim[1], Sanghoon Jeon[2*]
[1]Student, Department of Computer Software, The University of Suwon
[2]Professor, Department of Information and Security, The University of Suwon

요 약 공공장소에 Wi-Fi가 설치되는 사례가 증가함에 따라, 비밀번호가 없거나 다수가 동일한 비밀번호를 공유하는 네트워크 환경에서의 보안 위협이 심화되고 있다. 특히 Evil Twin 공격을 통해 사용자의 민감한 개인정보가 탈취될 수 있어, 이에 대한 보안 강화 방안이 요구된다. 본 논문에서는 사용자마다 서로 다른 비밀번호를 사용하도록 하여 Evil Twin 공격을 원천적으로 차단할 수 있는 방안으로, RADIUS 서버 기반의 랜덤 계정 생성 시스템(R-RAGS, RADIUS-based Random Account Generation System)을 제안한다. 제안된 시스템은 Spring Boot 기반의 웹 서버, RADIUS 서버, 그리고 MariaDB로 구성되며, 오픈소스로 구현되어 누구나 활용할 수 있도록 공개하였다. 실험을 통해 제안 시스템을 적용한 Wi-Fi 환경에서는 외부인이 유추할 수 없는 랜덤 비밀번호가 사용자마다 개별적으로 부여되기 때문에 Evil Twin 공격 수행이 사실상 불가능함을 확인하였다. 본 연구는 공공 Wi-Fi 보안성 강화를 위한 오픈코드 기반 접근을 제시하였으며, 제안 시스템이 다양한 공공장소에 적용되어 보다 안전한 무선 네트워크 환경을 조성하는 데 기여할 것으로 기대한다.

주제어 : 와이파이 보안, 공공 와이파이, RADIUS, Evil Twin 공격, 보안

Abstract As the installation of Wi-Fi in public spaces becomes increasingly common, security threats are intensifying in network environments where no password is set or a shared password is used by many users. In particular, since sensitive personal information of users can be stolen through Evil Twin attacks, security enhancement measures are required. In this paper, we propose a RADIUS server-based random account generation system (R-RAGS, RADIUS-based Random Account Generation System) as a method to fundamentally block Evil Twin attacks by allowing each user to use a different password. The proposed system consists of a Spring Boot-based web server, a RADIUS server, and MariaDB, and is implemented as open source so that anyone can use it. Through experiments, we confirmed that Evil Twin attacks are virtually impossible in Wi-Fi environments where the proposed system is applied because random passwords that cannot be guessed by outsiders are individually assigned to each user. This study presents an open code-based approach to enhance public Wi-Fi security, and we expect that the proposed system will be applied to various public places to contribute to creating a safer wireless network environment.

Key Words : Wi-Fi Security, Public Wi-Fi, RADIUS, Evil Twin Attack, Security

본 연구는 원자력안전위원회의 재원으로 한국원자력안전재단의 지원을 받아 수행한 원자력안전연구사업의 연구결과입니다 (No. RS-2024-00403596).
*교신저자 : 전상훈(shjeon@suwon.ac.kr)
접수일: 2025년 03월 10일 수정일: 2025년 03월 31일 심사완료일: 2025년 04월 07일

1. 서론

식당, 카페 등의 공공장소에서, 공용 Wi-Fi를 사용하는 사용자가 증가하고 있으나, 이들 중 상당수는 공용 Wi-Fi의 위험성을 인지하지 못하고 있다[1, 2].

현재 여러 공공장소의 Wi-Fi는 WPA(Wi-Fi Protected Access)2-PSK(Pre-Shared Key) 보안 프로토콜을 사용하고 있다[3]. 이러한 Wi-Fi를 대상으로는, 스니핑, ARP 스푸핑 공격, Evil Twin 공격 등을 수행할 수 있다[4]. 그중 Evil Twin 공격은, 정상적인 Wi-Fi 명칭과 동일한 Wi-Fi를 만든 후, 사용자가 공격자의 Wi-Fi 네트워크에 접속하도록 한 후 공격을 수행하는 것을 말한다[5]. 공격자는 이 공격을 활용하여 중간에서 접속자를 피싱 사이트로 우회시켜 비밀번호 등의 민감한 개인정보를 탈취할 수 있다[6].

Evil Twin 공격을 효과적으로 방지하기 위해서는 모든 사용자가 Wi-Fi 접속 시 각기 다른 비밀번호를 사용하는 방식이 요구된다. 이를 위해 RADIUS(Remote Authentication Dial-In User Service) 서버를 활용하면, Wi-Fi 네트워크에 여러 사용자의 정보 및 비밀번호를 등록하고 사용자별로 개별 비밀번호를 설정할 수 있다[7]. 이와 같은 접근 방식으로 공용 Wi-Fi 환경에서 Evil Twin 공격을 방지할 수 있다.

본 연구의 목적은 랜덤 문자열로 구성된 RADIUS 계정을 자동으로 생성하는 소프트웨어를 개발하고, 이를 RADIUS 서버와 연동하여 Evil Twin 공격을 방지할 수 있는 시스템인 R-RAGS(RADIUS-based Random Account Generation System)를 제안한다. 제안한 시스템은 저비용 하드웨어와 오픈소스를 활용하여 누구나 쉽게 구축할 수 있도록 설계되었다. 이를 통해, 공공 Wi-Fi 보안의 접근성을 높이고 보다 안전한 네트워크 환경을 조성하는 것을 목표로 한다.

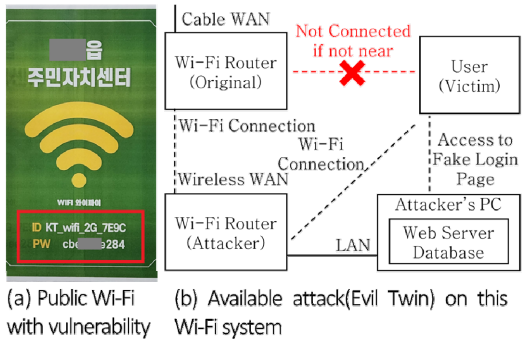
또한 본 연구에서는, 제안된 R-RAGS 시스템이 적용된 Wi-Fi 환경과 일반 Wi-Fi 환경의 보안 성능을 실험적으로 분석하고, Evil Twin 공격에 대한 방어 효과를 검증하였다.

본 논문의 구성은 다음과 같다. 2장에서는 Wi-Fi 공격 모델(Evil Twin)에 대해 다루며, 3장에서는 제안하는 R-RAGS 시스템의 설계 및 구현 방법을 제시한다. 4장에서는 제안 시스템의 실제 적용 사례 및 보안 성능 비교 실험을 다루며, 5장에서는 본 연구의 결론과 함께 향후 연구 방향을 제시한다.

2. 공격 모델

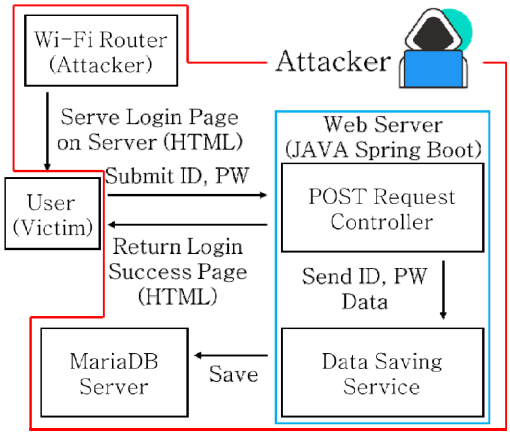
2.1 Evil Twin 공격 절차

Fig. 1(a)는 현재 대다수의 공공장소에서 흔히 볼 수 있는 WiFi 접근 방식을 나타낸다. 그림과 같이, 입구 또는 카운터 등의 장소에 Wi-Fi의 명칭(SSID, Service Set Identifier)과 비밀번호가 쓰인 종이가 부착하여 사용자에게 네트워크 정보를 제공하는 방식이다. 이러한 방식은 사용자 편의성 측면에서는 유리하지만, 보안 측면에서는 민감한 정보가 쉽게 노출될 수 있다는 단점을 가진다. Fig. 1(b)는 해당 방식의 취약점을 이용한 Evil Twin 공격의 과정을 도식화한 것이다. 공격자는 실제 네트워크와 동일한 SSID와 비밀번호를 가진 공유기를 설치하고, 사용자(공격 대상)가 이를 정상 네트워크로 오인하여 공격자의 Wi-Fi에 접속하도록 유도한다. 사용자가 공격자의 Wi-Fi에 접속하게 되면, 공격자는 트래픽을 가로채거나 피싱 사이트로 리디렉션시켜 아이디, 비밀번호 등과 같은 민감한 정보를 탈취하는 공격을 수행할 수 있다. 본 연구에서는 유명 사이트의 인증 화면과 동일한 환경을 구현하고 아이디와 비밀번호를 탈취하는 공격인 피싱 공격을 수행한다[8].



[Fig. 1] Evil Twin Attack

Fig. 2는, Evil Twin 공격을 이용한 피싱 공격 모델을 나타낸 것이다. 사용자(User, 공격 대상)가 공격자의 네트워크에 접속하면, 자동으로 공격자의 웹서버에 호스팅된 로그인 페이지(피싱 페이지)로 리디렉션된다. 사용자가 해당 페이지에 아이디와 비밀번호를 입력하고 전송하면, 입력된 정보는 공격자의 데이터베이스(MariaDB Server)에 저장되며, 이로 인해 민감한 인증 정보가 탈취될 위험이 있다.



[Fig. 2] Evil Twin Phishing Attack Model (Icon image: Korea Copyright Commission)

2.2 공격용 소프트웨어

공격에 사용된 웹 서버 소프트웨어(웹서버)는 Spring Boot 3.4.3을 활용하여 Java 언어로 구현되었다. Spring Boot는 복잡한 설정 없이 효율적으로 웹 어플리케이션을 개발하고자 할 때 사용되는 웹 프레임워크이다[9]. 사용자가 특정한 웹 주소 (/api/create/CollectedData)로 아이디와 비밀번호(개인정보)를 POST 요청으로 보내면, 해당 정보가 공격자의 데이터베이스에 저장된다.

Fig. 3는 공격용 웹서버의 컨트롤러 클래스 코드의 일부를 나타낸 것이다. 컨트롤러는 사용자로부터 요청이 들어오면 작업을 수행하는 서비스를 호출하고, 그 결과를 사용자에게 반환하는 역할을 수행한다[10]. 이 컨트롤러는 POST 요청을 통해 들어온 값을 2개의 문자열이 저장될 수 있는 CollectedDataForm 객체로 만들고, 그 객체를 collectedDataService(데이터베이스에 값을 저장해주는 서비스 객체)로 전달한다. 이후 서버는 사용자를 '로그인 성공' 문구가 표시되는 페이지로 리디렉션함으로써, 사용자가 정상적인 로그인 과정이라고 인식하도록 위장한다.

```
@PostMapping("/api/create/CollectedData")
public RedirectView create
        (@ModelAttribute CollectedDataForm collectedDataForm){
    collectedDataService.create(collectedDataForm);
    return new RedirectView( url: "/available.html");
}
```

[Fig. 3] Part of controller's code for Evil Twin Attack back-end server

Fig. 4는 피싱 공격에 사용된 HTML 페이지의 일부 코드를 나타낸 것이다. 이 페이지는 사용자로부터 아이디와 비밀번호를 입력받기 위한 폼(form)으로 구성되어 있으며, 사용자가 입력을 완료한 후 '전송' 버튼을 클릭하면, 앞서 Fig. 3에서 설명한 컨트롤러에 의해 해당 데이터가 POST 방식으로 서버에 전달된다. 이후 정보는 공격자의 데이터베이스에 저장된다. 해당 페이지는 실제 로그인 페이지와 유사한 형태로 설계되어, 사용자가 의심 없이 정보를 입력하도록 유도한다.

```
<form action="/api/create/CollectedData" method="post"><ul>
    <li><input type="text" name="username" placeholder="아이디" title="아이디"></li>
    <li><input type="password" name="password" placeholder="비밀번호" title="비밀번호"></li>
    <li><input type="submit" value="로그인"></li>
</ul></form>
```

[Fig. 4] HTML code for Evil Twin Attack front-end page

3. 제안 방법

Evil Twin은 다수가 동일한 공개 비밀번호를 공유하는 환경에서 발생할 수 있는 보안 위협으로, 공격자가 유사한 네트워크 환경을 위조하여 사용자가 위장된 네트워크에 쉽게 접속하도록 유도하는 방식이다. 이러한 공격을 방지하기 위해서는 사용자마다 서로 다른 인증 정보를 사용하는 방식이 요구된다.

본 연구에서는 사용자별 개별 인증을 지원하는 RADIUS 서버를 기반으로, 이를 연동한 시스템인 R-RAGS를 제안한다. RADIUS 서버는 R-RAGS의 핵심 구성 요소로 사용된다.

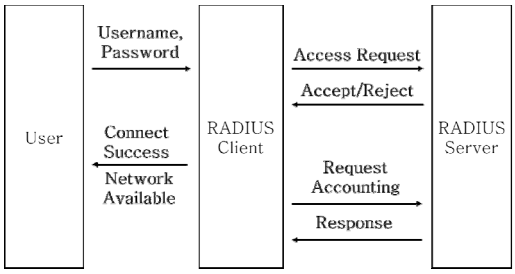
3.1장에서 RADIUS에 프로토콜의 개요 및 동작 방식에 대해 설명하고, 이어지는 3.2장에서는 제안하는 RADIUS 랜덤 계정 생성 시스템인 R-RAGS의 구조와 구현 방식을 상세히 기술한다.

3.1 RADIUS 설명

RADIUS 서버는 Wi-Fi 네트워크에 연결을 시도하는 사용자의 인증 정보(아이디, 비밀번호)를 바탕으로 신원을 확인하고, 네트워크 접근을 승인하는 역할을 수행한다.

Fig. 5는, RADIUS 서버의 인증(Authentication) 및 회계 처리(Accounting) 절차를 도식화한 것이다. 이 과정에서 Wi-Fi 공유기(AP)는 RADIUS 클라이언트의 역할을 수행한다[11]. 사용자가 Wi-Fi 공유기에 사용자명과 비밀번호를 입력하면, 공유기는 이를 RADIUS 서버로 전달하고, 서버는 해당 정보를 바탕으로 인증 절차를 수행한다. 인증 결과에 따라 서버는 성공(Accept) 또는 실패(failure) 메시지를 반환한다. 인증이 성공하면, 사용

자는 네트워크 요소에 접근할 수 있는 권한을 부여받는다[12]. 그 후에는 RADIUS 클라이언트와 서버가 회계 처리를 주기적으로 수행하여 상태 업데이트를 수행한다[7].



[Fig. 5] Working process of RADIUS

Fig. 6(a)와 Fig. 6(b)는 쉘 명령어를 이용하여 RADIUS 서버의 인증 동작을 직접 시험한 결과를 나타낸 것이다. Fig. 6(a)는 사용자 ID가 'testuser1'이고 비밀번호가 'testpass1'인 유효한 계정 정보를 전송한 경우로, RADIUS 서버로부터 인증이 승인되었음을 확인할 수 있다. 반면 Fig. 6(b)는 데이터베이스에 존재하지 않는 잘못된 계정 정보를 전송한 경우로, 인증 요청이 거부된 결과를 보여준다. 이를 통해, 제안한 시스템에서 RADIUS 서버가 사용자 인증 정보를 기반으로 정확한 접근 제어를 수행함을 확인 할 수 있다.

```
pi@pi:~ $ radtest testuser1 testpass1 localhost 0 1234
Sent Access-Request Id 106 from 0.0.0.0:38276 to 127.0
        User-Name = "testuser1"
        User-Password = "testpass1"
        NAS-IP-Address = 127.0.1.1
        NAS-Port = 0
        Message-Authenticator = 0x00
        Cleartext-Password = "testpass1"
Received Access-Accept Id 106 from 127.0.0.1:1812 to 1
pi@pi:~ $
```

(a) Valid account credentials sent

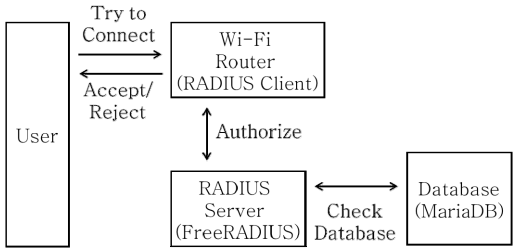
```
pi@pi:~ $ radtest testuser1 testpass2 localhost 0 1234
Sent Access-Request Id 214 from 0.0.0.0:52007 to 127.0.
        User-Name = "testuser1"
        User-Password = "testpass2"
        NAS-IP-Address = 127.0.1.1
        NAS-Port = 0
        Message-Authenticator = 0x00
        Cleartext-Password = "testpass2"
Received Access-Reject Id 214 from 127.0.0.1:1812 to 12
(0) -: Expected Access-Accept got Access-Reject
pi@pi:~ $
```

(b) Invalid account credentials sent

[Fig. 6] Authorizing account using shell

Fig. 7은 데이터베이스와 연결된 RADIUS 서버를 활용하여 사용자가 Wi-Fi 네트워크에 연결하는 전체 과정을 도식화한 것이다. 사용자가 스마트폰 등의 기기의 Wi-Fi 설정 메뉴 등을 이용하여 네트워크에 연결을 시도하면, 입력한 아이디와 비밀번호는 RADIUS 클라이언트에 해당되는 공유기를 통해 RADIUS 서버로 전달된다.

RADIUS 서버는 연결된 데이터베이스를 참조하여 해당 계정 정보의 유효성을 검증하고, 그 결과(승인 또는 거절)를 클라이언트에 반환한다. 인증이 승인되면 사용자는 인터넷에 정상적으로 접속할 수 있으며, 인증이 실패할 경우 네트워크 연결은 거부된다.



[Fig. 7] Connection process of RADIUS system connected to database

3.2 제안 시스템

3.2.1 시스템의 작동 방식

본 논문에서는 랜덤한 문자열을 기반으로 RADIUS 계정을 자동으로 생성하고 사용자에게 제공하는 시스템인 R-RAGS(RADIUS-based Random Account Generation System)를 제안한다. 이 시스템은 사용자가 웹 서버에게 계정 생성을 요청하면, 무작위로 생성된 두 개의 문자열을 각각 아이디와 비밀번호로 사용하여, 해당 정보를 RADIUS가 참조하는 MariaDB에 저장한 후, 이를 사용자에게 보여주는 방식으로 작동된다.

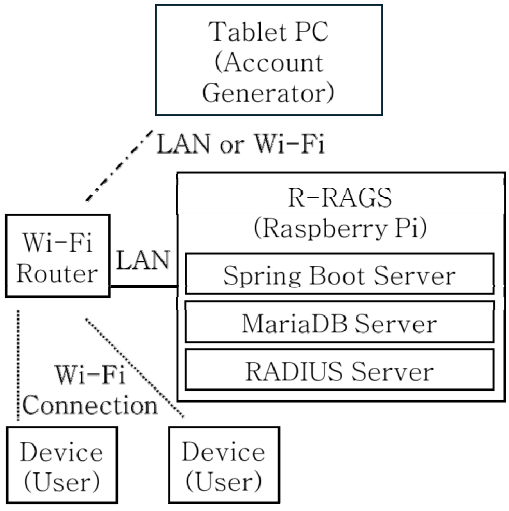
백엔드 웹 서버는 Java 언어와 Spring Boot 3.4.3 프레임워크를 기반으로 구현되었다. 본 연구는 공공 Wi-Fi 환경의 보안성 강화를 위한 오픈코드 취지에 따라, 개발된 소스코드를 오픈소스 플랫폼(GitHub)을 통해 공개하였다[1].

Fig. 8은 제안하는 R-RAGS 시스템의 전체 구조를 도식화한 것이다. R-RAGS는 Spring Boot 기반의 웹 서버, MariaDB 데이터베이스 서버, RADIUS 인증 서버로 구성되며, 이들은 Raspberry Pi에 설치되어 통합적으로 동작한다.

Spring Boot 서버에는 프론트엔드 페이지가 함께 포

1) https://github.com/mgkim1/r_rags_web_server

함되어 있으며, 해당 페이지는 태블릿 PC를 통해 사용자에게 표시된다. 계정 생성 시, 생성된 계정 정보는 MariaDB 데이터베이스에 저장된다. 사용자가 Wi-Fi 네트워크에 연결을 시도할 경우, 사용자가 Wi-Fi에 연결할 때는 사용자가 입력한 아이디 및 비밀번호 정보와 MariaDB에 저장된 데이터를 RADIUS 서버가 비교하여 인증을 수행한다. 인증이 성공하면 사용자는 네트워크에 정상적으로 접속할 수 있다.



[Fig. 8] System overview of RADIUS-based Random Account Generation System

3.2.2 프론트엔드 페이지 구성

Fig. 9은 R-RAGS의 프론트엔드 웹페이지를 나타낸 것이다. 해당 페이지는 공공장소의 로비나 카운터 등 사용자 접근이 용이한 위치에 설치한 태블릿 PC를 통해 제공되며, 화면이 꺼지지 않도록 설정하여 지속적으로 표시된다. 사용자가 화면의 'Wi-Fi 비밀번호 보기' 버튼을 터치하면 /newAccount URL로 이동하여 새로운 계정 정보(아이디 및 비밀번호)를 생성하고 이를 사용자에게 시각적으로 제공한다.

[Fig. 9] Front-end main page of RADIUS-based Random Account Generation System

Fig. 10은 프론트엔드 페이지에서 'Wi-Fi 비밀번호 보기' 버튼을 터치하였을 때 표시되는, 백엔드에 의해 랜덤하게 생성된 계정 정보(아이디 및 비밀번호)가 화면에 표시되는 모습을 나타낸다. 사용자는 이 계정 정보를 이용하여 Wi-Fi에 로그인할 수 있다.

화면 하단에 배치된 '연결 완료 후 이 버튼을 눌러주세요' 버튼은 시각적으로 강조되도록 디자인되었으며, 이는 모든 사용자가 고유한 계정을 사용하도록 유도하기 위한 목적으로 구성되었다. 사용자가 Wi-Fi 연결 후 이 버튼을 누르지 않을 경우, 동일한 계정 정보가 화면에 지속적으로 표시되어 다음 사용자에게 재사용될 가능성이 있기 때문에, 계정 공유를 방지하기 위한 조치로 해당 버튼 클릭을 유도한다..

이 버튼을 클릭하면 Fig. 9의 메인페이지로 이동되며, 이후 다시 'Wi-Fi 비밀번호 보기' 버튼을 터치할 경우, 새로운 계정 정보가 생성되어 화면에 표시된다.

[Fig. 10] Screen appeared when ID(username) and password is generated

3.2.3 백엔드 서버의 구성

Spring Boot(Java)를 활용하여 구현된 백엔드 서버는 Entity(클래스명 Radcheck), Repository, Service, Controller로 구성된다.

Fig. 11은 컨트롤러 코드의 일부를 나타낸 것이다. 사용자가 /newAccount 주소에 접근하여 GET 요청을 하면, Controller는 Service를 호출하여 새로운 계정 정보를 생성하도록 한다. 이후, 생성된 계정 정보를 조회할 수 있는 페이지인 그림 10과 같은 newAccountPage로 사용자를 이동시킨다. 이 페이지에는 Service가 반환한 객체에 저장된 랜덤한 문자열(아이디, 비밀번호 정보)이 표시되어 사용자가 Wi-Fi 접속에 사용할 수 있도록 한다.

```
@GetMapping("/newAccount")
public String createRadiusAccount(Model model) {
    model.addAttribute( attributeName: "account", radcheckService.create());
    return "newAccountPage"; // newAccountPage.html로 이동
}
```

[Fig. 11] Part of Controller's Code for RADIUS Account Generator Back-end

Fig. 12는 서비스 클래스 코드의 일부를 나타낸 것이다. 해당 클래스에서는 generateRandomString() 메서드를 통해 각각 12자의 랜덤 문자열 두 개를 생성하며, 이를 각각 아이디와 비밀번호로 사용한다. 생성된 문자열을 기반으로 Radcheck Entity 객체를 생성하고, 이를 Repository 객체에 전달하여 데이터베이스에 저장한다. 저장이 완료된 Entity 객체는 호출한 컨트롤러(Controller)로 반환되어, 이후 사용자에게 계정 정보로 제공된다.

```
public Radcheck create() {  1 usage
    // 랜덤한 아이디, 비밀번호로 계정을 만들어서
    // DB에 저장하고, 계정 정보를 반환
    String username = generateRandomString(); // 사용자명 랜덤생성
    while(radcheckRepository.existsByUsername(username))
        username = generateRandomString();
    // 만약 아이디가 중복되었다면 다시 생성
    Radcheck radcheck = new Radcheck
            ( id: null, username, attribute: "Cleartext-Password",
                    op: ":=", value: generateRandomString());
    // 생성된 (중복되지 않는) 아이디와,
    // 랜덤생성된 비밀번호로 Radcheck 객체 만들기
    radcheckRepository.save( entity: radcheck);
    // 만들어진 radcheck 객체를 db에 저장
    return radcheck; // 만들어진 객체(아이디, 비번 담겨있는) 반환
}
```

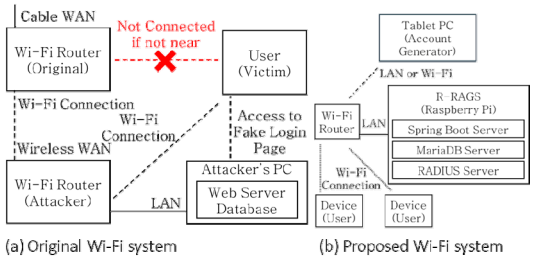
[Fig. 12] Part of service's code for RADIUS's back-end server

4. 실험 결과

4.1 실험 환경

본 실험은 본 연구에서 제안한 R-RAGS 시스템의 효과를 검증하기 위해, R-RAGS가 적용되지 않은 기존 Wi-Fi 환경과 제안된 시스템이 적용된 Wi-Fi 환경을 비교하는 방식으로 수행되었다.

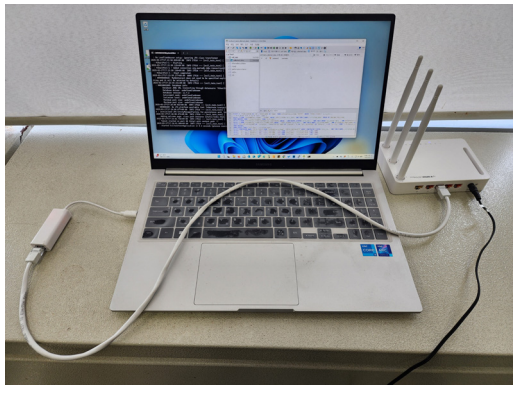
R-RAGS가 적용되지 않은 경우는 Fig. 13(a) 구조로 구성하여 실험을 수행하였으며, R-RAGS가 적용된 경우는 Fig. 13(b)의 구조를 기반으로 시스템을 구축하고 실험을 진행하였다.



[Fig. 13] Research environment

기존 시스템은 두 개의 Wi-Fi 공유기(정상 공유기 및 공격자 공유기), 공격자용 컴퓨터, 그리고 네트워크 사용자용 휴대폰으로 구성된다.

Fig. 14는 해당 실험에서 사용된 공격자 컴퓨터를 나타내며, Windows 11 Home 운영체제를 기반으로 Intel i5-1340P(1.90 GHz) 프로세서, 16GB RAM, Liberica JDK 21이 설치되어 있다. 이 컴퓨터에서는 제2장에서 개발한 피싱 사이트를 제공하는 웹 서버가 구동되며, 사용자로부터 수집된 정보를 저장하기 위한 데이터베이스로는 MariaDB 11.7.2.0이 설치되어 운영된다. 한편, 네트워크 사용자 단말기는 Android 14 운영체제를 기반으로 하며, Qualcomm Snapdragon 8 Gen 1 프로세서와 12GB RAM을 탑재하고 있다.



[Fig. 14] Computer for conducting Evil Twin Attack

제안하는 시스템은 Evil Twin 실험과 달리, 정상 공유기만 존재하며 공격자 공유기는 포함되지 않는다. 일반적으로 Evil Twin 공격을 수행하기 위해서는 사용자가 실제로 사용하는 비밀번호를 공격자가 알고 있어야 하며, 이를 기반으로 동일한 SSID(Service Set Identifier)를 가진 가짜 네트워크를 구축해야 한다.

그러나 제안한 시스템에서는 사용자마다 랜덤하게 생성된 계정 정보가 데이터베이스에 저장되므로, 이를 알기 위해서는 해당 정보를 보관하고 있는 Raspberry Pi 장비에 직접 접근해야 한다. 이러한 구조적 특성으로 인해, 제안된 시스템 구조에서는 Evil Twin 공격 수행이 사실상 불가능하다.

제안 시스템은 공유기 외에도 서버 역할을 수행하는 Raspberry Pi 5 장비(OS: Raspberry Pi OS, CPU: ARM Cortex-A76, RAM: 8GB)와, 계정 정보를 사용자에게 표시하기 위한 태블릿 PC(OS: Android 11, CPU: Qualcomm Snapdragon 429, RAM: 2GB)로 구성된다. R-RAGS가 설치되는 컴퓨터로 Raspberry Pi를 선정한 이유는, 소형 폼팩터로 인해 공유기와 함께 설치하기 용이하고, 저렴한 가격으로 장비 설치에 대한 경제적 부담이 적기 때문이다[13]. 기존 시스템과 제안 시스템의 장치 구성 차이는 Table 1에 정리되어 있다.

〈Table 1〉 Difference of Experimental Setup

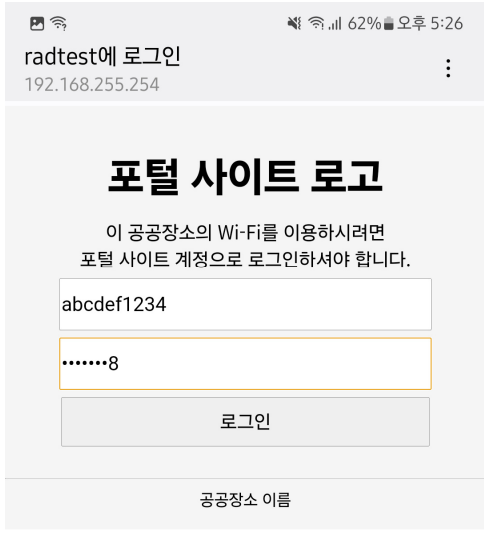
Device \ System	Original system	Proposed system
Original Router	Used	Used
Attacker's Router	Used	Not Used
Attacker's Computer	Used	Not Used
Raspberry Pi(R-RAGS)	Not Used	Used
Tablet Computer	Not Used	Used
Phone for Wi-Fi User	Used	Used

4.2 기존 시스템의 보안성 검증

R-RAGS가 적용되지 않은 Wi-Fi 시스템의 보안성을 실험을 통해 검증하였다. 공격자는 공격용 공유기에 자신의 컴퓨터를 연결하고, 해당 컴퓨터에는 피싱 페이지 출력을 위한 웹 서버와 수집된 정보를 저장할 MariaDB가 설치되어 있다. 이후 공유기에 내장된 '공지/광고 기능' 설정을 활성화하였다. 이 기능은 네트워크에 연결된 장치가 설정된 URL에 강제로 접속하도록 하는 기능이다[14]. 본 실험에서는 사용자가 공격자의 네트워크에 접속할 경우, 자동으로 공격자의 웹 서버 페이지로 연결되도록 설정하였다.

실험 결과, 정상 공유기 근처에서 연결을 시도하면 정상 공유기로 연결되어 피싱 페이지를 확인할 수 없다. 반면, 공격자의 공유기 주변에서 연결을 시도하면 공격자의 공유기로 연결되어 피싱 페이지로 연결되는 로그인 알림이 화면에 표시된다.

Fig. 15은 공격자의 네트워크에 연결된 후, 표시되는 로그인 알림을 터치하면 연결되는 로그인 페이지의 화면을 보여준다. 이 페이지는 공격자의 웹 서버에서 제공되는 피싱 페이지이다. 이를 피싱 페이지임을 인지하지 못한 사용자가 아이디와 비밀번호를 입력하고 전송할 경우, 해당 개인정보가 공격자의 서버로 전송되어 탈취될 수 있다.

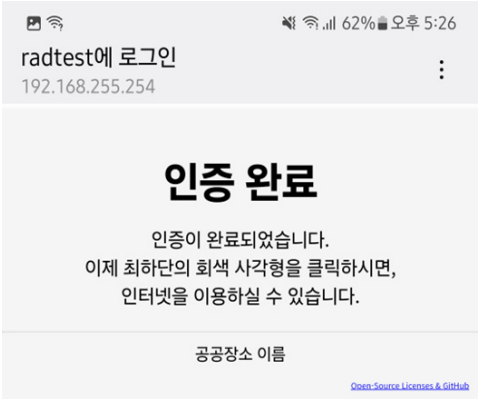


[Fig. 15] Screen of mobile device connected to attacker's network

Fig. 16은 공격자 컴퓨터에 구축된 데이터베이스의 일부를 보여준다. 사용자가 피싱 페이지에 아이디와 비밀번호를 입력하여 전송하면, 해당 정보는 웹 서버를 통해 수신되며, 이후 자동으로 공격자 컴퓨터의 MariaDB에 저장된다. 이렇게 수집된 데이터는 공격자가 사용자 개인정보를 탈취하는 데 활용될 수 있다.

[Fig. 16] Collected data saved in database on attacker's computer

Fig. 17은 사용자가 로그인 버튼을 클릭한 후, 공격 대상 기기에 표시되는 화면을 보여준다. 입력된 값의 유효성과 무관하게, 항상 동일한 화면이 출력되도록 설계되어 있다. 이는 사용자가 정상적인 인증이 이루어진 것으로 오인하도록 유도하기 위한 기법이다.



[Fig. 17] Screen of mobile device after clicking login button

위 실험을 통해, 동일한 비밀번호가 모든 사용자에게 공유되는 Wi-Fi 환경에서는 Evil Twin 공격이 실제로 수행 가능함을 확인할 수 있었다. 이는 공용 비밀번호 기반 인증 방식의 구조적 취약점을 보여주는 결과로, 사용자 개별 인증이 이루어지지 않는 환경에서는 보안 위협에 쉽게 노출될 수 있음을 보여준다.

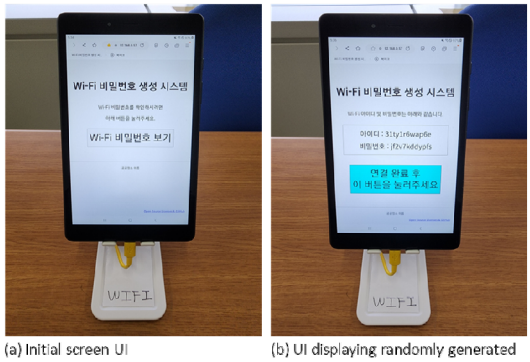
4.3 제안 시스템의 보안성 검증

Fig. 18은 제안 시스템의 구성 방식을 나타낸 것으로, Raspberry Pi 장비를 Wi-Fi 공유기에 연결한 후 해당 장비에 제안 시스템의 구성 요소를 설치한 형태이다.

구체적으로는, 사용자 계정 정보를 저장하기 위한 MariaDB 서버, Wi-Fi 접속 시 인증 절차를 수행하는 FreeRADIUS 서버, 그리고 계정 정보를 랜덤으로 생성하고 사용자에게 표시하는 역할을 수행하는 Spring Boot 기반의 웹 서버가 설치된다. 본 시스템에서 RADIUS 서버로 FreeRADIUS를 사용하는 이유는 이것이 리눅스에서 사용 가능한 오픈소스 기반 RADIUS 서버 소프트웨어이기 때문이다[15]. 위 세 가지 구성 요소는 R-RAGS를 구성하는 핵심 요소이다.

[Fig. 18] Raspberry Pi connected with Wi-Fi router

Raspberry Pi를 이용하여 Wi-Fi 접속용 계정 1개를 생성한 후, 터치스크린이 포함된 기기에 해당 계정을 이용해 Wi-Fi에 접속시킨다. 이후, 해당 기기를 Fig. 19와 같이 공공장소의 카운터나 출입구 등 사용자의 접근성이 높은 위치에 설치한다. 이를 통해, 사용자는 언제든지 화면을 통해 새로운 계정 정보를 발급받을 수 있으며, Wi-Fi 보안성 또한 유지할 수 있다.



(a) Initial screen UI (b) UI displaying randomly generated account credentials

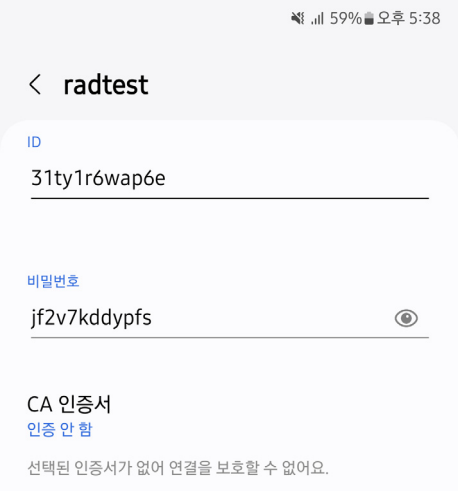
[Fig. 19] Tablet computer with display of R-RAGS's Front-end

Fig. 20는 Raspberry Pi 컴퓨터의 RADIUS 계정 데이터베이스의 일부를 나타낸 것이다. 이 데이터베이스에는 Fig. 10 및 Fig. 19에서 생성된 계정을 포함하여, 웹 서버를 통해 생성된 모든 사용자 계정 정보가 저장된다. 사용자가 웹사이트를 통해 추가로 계정 정보를 발급받는다면, 발급된 랜덤 아이디 및 비밀번호가 이 데이터베이스에 추가로 저장된다. 이후 Wi-Fi 인증 요청이 발생하면, RADIUS 서버는 이 데이터베이스를 참조하여 해당 계정의 존재 여부를 확인하고, 그 결과에 따라 인증을 승인하거나 거절하게 된다.

```
MariaDB [radius]> SELECT * FROM radcheck;
+----+-------------+--------------------+----+-------------+
| id | username    | attribute          | op | value       |
+----+-------------+--------------------+----+-------------+
|  1 | szirhlkr3kc6 | Cleartext-Password | := | 3sa8nz5u6i3r |
|  2 | drag8bbq9qq8 | Cleartext-Password | := | i4cam8js8z4z |
|  3 | plsg7a7g867x | Cleartext-Password | := | qaiwtc08hiuo |
|  4 | qsgdrv2jq5zs | Cleartext-Password | := | hlltevz317s4 |
|  5 | a8vrgec5z837 | Cleartext-Password | := | yl91jajxn738 |
|  6 | y8tpgegkorf2 | Cleartext-Password | := | cym4rm8lnuh2 |
|  7 | dfojc41dqvlp | Cleartext-Password | := | uzh7jkovos7u |
|  8 | 4nhh6h3tv4w6 | Cleartext-Password | := | h4hht1vyzzcp |
|  9 | hyioz8jsikwy | Cleartext-Password | := | rk3jjonzrtq5 |
| 10 | n6kug0bl1hp6 | Cleartext-Password | := | eyy3bicjjcbo |
| 11 | plowtms4doem | Cleartext-Password | := | 5q9lxk57z5go |
| 12 | 31ty1r6wap6e | Cleartext-Password | := | jf2v7kddypfs |
+----+-------------+--------------------+----+-------------+
12 rows in set (0.001 sec)
```

[Fig. 20] Randomly generated account data in database

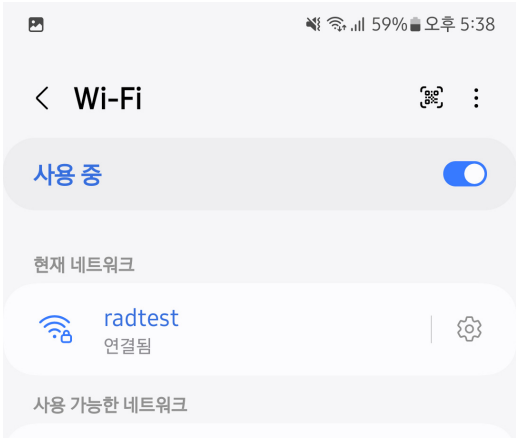
Fig. 21은 모바일 기기를 이용하여 Fig. 19에 표시된 계정 정보로 로그인을 시도하는 장면을 보여준다. 해당 아이디 및 비밀번호는 데이터베이스에 이미 등록되어 있으므로, 네트워크 연결에 성공하게 된다.



[Fig. 21] Information for Wi-Fi Connection

Fig. 22는 올바른 계정 정보를 입력하여 인증에 성공한 후, 모바일 기기가 Wi-Fi 네트워크에 정상적으로 연결된 상태를 보여준다. 반대로, 잘못된 아이디 또는 비밀번호를 입력할 경우 RADIUS 서버에서 인증이 거부되며, 네트워크에 접속할 수 없다.

제안된 시스템을 사용할 경우, 공격자가 R-RAGS의 데이터베이스에 저장된 계정 정보를 직접 획득하지 않는 이상, 해당 네트워크를 대상으로 Evil Twin 공격을 수행하는 것은 사실상 불가능하다. 이는 Evil Twin 공격이 네트워크 사용자 각각의 개별 계정 정보를 사전에 알고 있어야만 성공할 수 있기 때문이다.



[Fig. 22] After authenticating Wi-Fi account

5. 결론

본 연구에서는 비밀번호가 있는 기존의 Wi-Fi 네트워크가 가진 Evil Twin 공격에 대한 취약성을 해결하는 방안을 제시하였다. RADIUS 인증 서버를 활용하여 제안한 RADIUS 랜덤 계정 생성 시스템(R-RAGS)을 통해, 사용자마다 고유한 아이디와 비밀번호를 부여받는 구조를 구현함으로써, Evil Twin 공격을 효과적으로 차단할 수 있음을 실험을 통해 확인하였다.

제안된 시스템은 RADIUS 기반 보안 구조를 고가의 인프라 없이도 구성할 수 있도록 설계되었으며, 기업이나 학교 등 대규모 기관이 아닌 소형 공공장소에서도 저비용 하드웨어를 활용해 보안성을 개선할 수 있다는 가능성을 제시한다. 또한, 해당 시스템은 오픈소스로 공개되어 있어, 누구나 자유롭게 설치하고 활용할 수 있으며, 이를 통해 다양한 공공장소의 Wi-Fi 환경에서 보안을 강화할 수 있을 것으로 기대된다.

향후 연구로는 본 논문에서 다루지 못한 ARP 스푸핑 등 기타 Wi-Fi 공격에 대한 대응 방안 개발이 필요하며, 소상공인 및 일반 사용자들도 손쉽게 본 시스템을 구축할 수 있도록 설치 절차의 단순화 및 자동화에 대한 연구를 수행할 계획이다.

결론적으로, 본 연구는 저비용의 하드웨어 기반으로 공공장소 Wi-Fi의 보안성을 향상시킬 수 있는 실용적인 대안을 제시하였으며, 제안된 시스템이 다양한 공공 환경에 확산되어 더 안전한 네트워크 사용 환경을 제공하는 데 기여할 것으로 기대된다.

REFERENCES

[1] Suzan Ali, Tousif Osman, Mohammad Mannan and Amr Youssef, "On Privacy Risks of Public WiFi Captive Portals," Data Privacy Management, Cryptocurrencies and Blockchain Technology: ESORICS 2019 International Workshops, DPM 2019 and CBT 2019, Luxembourg, September 26–27, 2019 Proceedings, pp.80-98, Sep. 2019.

[2] Hee-Ra Bae, Min-Young Kim, Su-Kyung Song, Seul-Gi Lee, and Younghyun Chang, "Security Attack Analysis for Wireless Router and Free Wi-Fi Hacking Solutions," The journal of the convergence on culture technology, Vol.2, No.4, pp.65-70, Dec. 2016.

[3] Byoungcheon Lee, "Efficient Wi-Fi Security Protocol Using Dual Tokens," Journal of The Korea Institute of Information Security & Cryptology, Vol.29, No.2, Apr. 2019.

[4] Dongmin Ahn, Youngil Kim, Younsung Choi and Dongho Won, "Analysis on Security Vulnerabilities of Wi-Fi and Response Model using Electronic Signature," Proceedings of the Korean Institute of Communication Sciences Conference, pp.367-368, Nov. 2014.

[5] Yeonsoo Jeong, Jonguk Kim, Sukin Kang and Manpyo Hong, "A Smart Phone Evil Twin Detection Method Using IP Address," Proceedings of the Korean Information Science Society Conference, Vol.39, No.1, pp.269-271, Jun. 2012.

[6] SungBae Kang, DaeHun Nyang and KyungHee Lee, "Evil-Twin Detection Scheme Using SVM with Multi-Factors," The Journal of Korean Institute of Communications and Information Sciences, Vol.40, No.2, pp.334-348, Feb. 2015.

[7] Kim, Young se, "Improved hash and transmission method for larger packets in the RADIUS protocol," Masters Thesis, Graduate School of Konkuk University, Feb. 2017.

[8] Hong Ryeol Ryu, Moses Hong and Taekyoung Kwon, "A Study of Multiple Password Leakage Factors Caused by Phishing and Pharming Attacks, " Journal of The Korea Institute of Information Security & Cryptology, Vol.23, No.6, Dec. 2013.

[9] Sung-Woo Jo, Jai-Soon Baek and Sung-Jin Kim, "CRM Customer Care Web Page for Personal Self-employment, " Proceedings of the Korean Society of Computer Information Conference, Vol.32, No.2, Jul. 2024.

[10] SunBeom Kwon, JaeYong Oh, SeungWoo Jo, SungJin Kim, HyungMook Lee and JunDong Lee, "Make Simple Blog with Spring Boot, " Proceedings of the Korean Society of Computer Information Conference, Vol.30, No.1, Jan. 2022.

[11] Lee Hae Dong, Choi Doo Ho and Kim Hyun Gon, "Mobile IPv6 Session Key Distribution Method At Radius-based AAAv6 System," Proceedings of the IEEK Conference, pp.581-584, Aug. 2004.

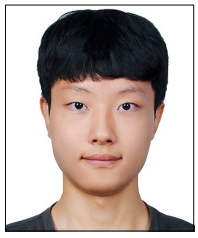
[12] Yeon-Woo Jeong, Jong-Yoon Sohn, Joong-Chang Chun and Kyung-Sun Choi, "Development of a RADIUS WLAN Security System for Industrial Applications Based on WEB," The Journal of Korea Institute of Information, Electronics, and Communication Technology, Vol.9, No.6, pp.599-603, Dec. 2016.

[13] Younggun Lee, Sanghyun Lee and Seunghoon Yoo, "Development of CanSat System based on Raspberry Pi, " Journal of The Institute of Electronics and Information Engineers, Vol.59, No.1, Jan. 2022.

[14] Dong-hun Kim, Min-sik Jeon, Ju-ho Choi, Jin-ho Choi and Dong-woo Lee, "Automated Installation of Softwares in Intranet Using Router Packet Analysis, " Proceedings of KIIT Conference, Jun. 2018.

[15] Young-Hoon Jin and Young-Yeol Choo, "P²HYMN: Hybrid Network Systems for Maintenance Support in Power Plants, " Journal of Institute of Control, Robotics and Systems, Vol.20, No.7, Jul. 2014.

김 명 규(Myeonggyu Kim) [준회원]



- 2023년 3월 ~ 현재 : 수원대학교 컴퓨터학부 컴퓨터SW 학과 학사 과정

〈관심분야〉
정보보안, SW개발, 사물인터넷, 스마트시티

전 상 훈(Sanghoon Jeon) [정회원]

- 2012년 2월 : 경북대학교 IT대학 심화 전자공학 공학사
- 2014년 2월 : 대구경북과학기술원 정보통신융합공학전공 공학석사
- 2020년 8월 : 대구경북과학기술원 정보통신융합전공 공학박사
- 2020년 3월 ~ 2020년 8월 : 한양대학교 산학협력단 선임연구원
- 2020년 9월 ~ 2022년 9월 : 한양대학교 의과대학 응급의학과 포닥연구원
- 2022년 10월 ~ 2023년 9월 : 한양대학교 의과대학 응급의학과 연구조교수
- 2023년 10월 ~ 현재 : 수원대학교 지능형SW융합대학 정보보호학과 조교수

〈관심분야〉
웨어러블컴퓨팅, 의료인공지능, CPS보안