

침해사고 예방을 위한 정보보안 교육훈련 문제은행 시스템

모은수* · 이재필* · 이재광* · 이준현* · 이재광*

*한남대학교

A Design of Information Security Education training Databank System for Preventing Computer Security incident

Eun-Su Mo* · Jae-Pil Lee* · Jae-Gwang Lee* · Jun-Hyeon Lee* · Jae-Kwang Lee*

*Hannam University

E-mail : {esmo · jplee · jglee · jhlee}@netwk.hnu.kr* · jklee@hnu.kr*

요 약

스미싱, 피싱 등의 개인정보 침해사고로 인한 개인정보보안이 화두가 되고 있다. 이와 같은 개인정보 침해 사건사고는 개인정보관리에 있어 사용자의 의식이 부족하기 때문에 발생한다. 본 논문에서는 기존의 XML Tag 구조 기반 문제은행 시스템과 달리 텍스트 기반 교환 형식 기술로 언어에 의존하지 않는 장점을 가진 Key-Value 방식의 JSON을 사용하였다. 제안하는 시스템은 정보보호 분야별 상, 중, 하의 난이도로 구분하며, 공간 및 시간 제약 없는 자유로운 스마트기기 및 PC를 통해 사용자에게 서비스를 제공한다. 교육훈련 서버(훈련서버)의 안정적인 서비스를 위하여 오픈소스 기반의 Nodejs와 Apache의 Load Balancing 기술을 사용한다. 또한 교육훈련의 정답, 오답 판정 시 훈련서버에게 요청하지 않고 웹페이지에서 처리하며, 그 결과는 jQuery Ajax를 이용하여 훈련서버에게 전송된다. 사용자 ID를 기준으로 데이터베이스에 저장되고, 교육훈련통계 지표로 사용하도록 하였다. 본 논문에서는 사용자의 정보보안 의식 강화를 위해 수준별 교육훈련 시스템을 설계하였다.

ABSTRACT

Smishing, Phishing personal privacy caused by Incident accidents such as Phishing information security has become a hot topic. Such incidents have privacy in personal information management occurs due to a lack of user awareness. This paper is based on the existing structure of the XML Tag question bank used a different Key-Value Structure-based JSON. JSON is an advantage that does not depend on the language in the text-based interchange format. The proposed system is divided into information security sector High, Middle and Low grade, and Provides service to the user through the free space and the smart device and the PC to the constraints of time. The use of open source Apache Load Balancing technology for reliable service. It also handles the user's web page without any training sessions Require server verification result of the training(training server). The result is sent to the training server using jQuery Ajax. and The resulting data are stored in the database based on the user ID. Also to be used as a training statistical indicators. In this paper, we design a level training system to enhance the user's information security awareness.

키워드

침해사고, 정보보안, 교육훈련, 문제은행

I. 서 론

최근 들어 개인정보 침해사고로 인한 스미싱, 피싱 등의 문제가 사회적 이슈가 되고 있다. 스미싱은 문자메세지(SMS)와 피싱(Phishing)의 합

성어로 스마트폰 사용자에게 악성 URL클릭으로 악성 앱을 설치하게 유도하여 개인정보 및 금융 정보를 탈취하는 기법중 하나이다[1]. KISA에서 발표한 자료에 따르면[그림1] 2014년 1월부터 9월

까지 악성코드 탐지건수는 631,490건이고 해킹사고 접수건수는 10,600건으로 악성코드를 이용한 해킹사고가 끊이지 않고 있다. 2013년 1월에서 9월까지의 해킹사고 접수 건은 10,600으로 전년 대비 7.21% 증가했다[2]. 하지만 탐지건수에 비해 신고건수가 크게 차이난다. 이것으로 보아 사용자들의 보안의식이 부족한 것을 알 수 있다.

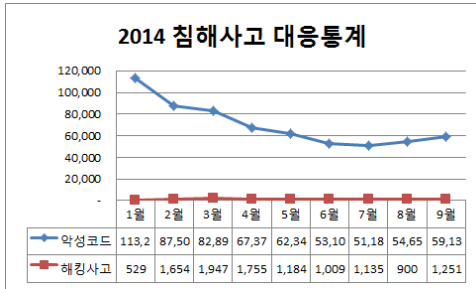


그림 1. 침해사고 대응통계[2]

그렇기 때문에 본 논문에서는 사용자 보안의식 강화를 위해 문제은행 방식의 교육훈련 시스템을 설계 및 구현 하였다. 제안하는 교육훈련 시스템은 정보보호 분야별 상, 중, 하의 난이도로 구분되며 공간 및 시간의 제약에 자유로운 모바일 기기 및 PC를 지원한다. 웹 표준을 지켜 다양한 브라우저에서도 지원하도록 하였다. 사용자는 자신이 취약한 분야를 테스트하며 즉시 확인 하여 재교육훈련을 받을 수 있도록 했다. 2장에서는 관련 연구를 소개하고 3장에서는 시스템을 설계한다. 4장에서는 시스템을 구현하며 결론과 향후 연구를 제시하며 마무리 한다.

II. 관련연구

2.1 기존 문제은행(item bank) 시스템

기존의 문제은행 시스템의 연구는 xml기반에서부터 시맨틱 웹기반 등 지속적으로 이뤄져 왔다 [3][4]. ‘문제은행방식’이란 ‘폐쇄형 출제방식’과는 달리 과목별로 충분한 양의 문제를 미리 축적해 두었다가 필요에 따라 문항들을 활용하는 제도를 말한다. 이는 양질의 문제를 생산할 수 있는 이점을 가지고 있다. 교통안전공단, 한국산업이력공단 등에서 이용하고 있다

2.2 Nodejs

Nodejs는 기존의 multi thread 기반의 서버와는 달리 single thread 기반으로 동작하는 고성능의 비동기 IO (Async / Non-blocking IO)를 지원하며 자체적으로 네트워크 서버를 구현할 수 있는 오픈소스 기반의 프로젝트 중에 하나이다. 이미 해외 기업인 아마존, LinkedIn, PayPal 등에서 사용중이며 국내에서는 쿠키런 등의 웹게임 개발에서

도 사용 중 이다. 기존의 웹 서비스는 blocking I/O 기반으로 클라이언트의 접속 시 마다 thread를 생성하여 CPU와 Memory 등의 서버의 사양이 중요하다. 이는 접속빈도가 증가 할수록 응답률이 저하되는 문제를 가지고 있기 때문이다[5]. 이를 제안하는 시스템에서는 cluster를 이용하여 서버의 증설 없이 대처 할 수 있도록 하였다.

2.3 express 프레임워크

express 프레임 워크는 Nodejs의 확장 모듈로 기본적인 뷰 지원과 세션을 지원한다. 설정기반구성이며 node자체적으로 http서버 구현 시 필요한 모듈과 기능을 이미 내장 하고 있어 개발에 용이하다는 장점을 가진다[6].

III. 시스템 설계

3.1 시스템 설계

안정적인 서비스를 위해 다수의 사용자 요청을 Load Balance통해 분산 처리 한다. 또한 Nodejs 자체적으로 Cluster를 이용하여 CPU의 개수만큼 worker를 생성하여 작업이 분산 된다. 사이트의 URL 처리는 express의 router 기능을 사용하였으며 빠른 응답을 위해 callback을 jQuery Ajax를 이용하여 페이지에서 JSON(JavaScript Object Notation) 데이터를 받아 처리하도록 했다.

동적 처리를 위해 페이지는JADE(JADE-Template Engine)을 사용된다. JADE는 클라이언트 사이드를 지원하고 간결하며 tag를 이용한 접두어 처리가 없는 것이 특징이다. AUM¹⁾기능을 위해 추가적으로 csv-parser, ssh2 등의 모듈이 사용되었다. 제안하는 시스템의 구성도는 [그림 2]와 같다.

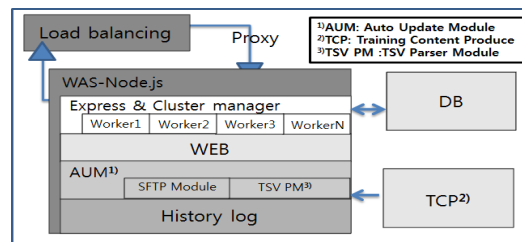


그림 2. 시스템 구성도

3.2 데이터베이스 설계

문제 테이블은 전체 문제를 관리하며 일련번호를 통해 관리된다[표 1]. 사용자 통계 데이터는 1회분 문제를 기준으로 수집되며 정답, 오답, 미응답수를 합산하여 총 문제 수를 파악 한다. 그리고 일시 데이터는 unix_timestamp를 이용하여 연산에 편의를 주었다[표 2]. 파일버전 관리를 위해 파일명은 yyyyymmdd.tsv 형식으로 생성되며 파일제공서버의 특정위치에 저장된다. 파일의 손상유무 및 백업을 위해 파일정보를 수집하게 된다[표3]

표 1. 문제 테이블

필드명	데이터 형식	비고
일련번호	int	식별번호
분야	varchar	관련분야
난이도	varchar	상 / 중 / 하
지문	text	
보기1	varchar	
보기2	varchar	
보기3	varchar	
보기4	varchar	
답안	int	정답표기 ex) 1
해설	text	문제 해설

표 2. 사용자 통계 테이블

필드명	데이터 형식	비고
일시	varchar	unix timestamp
사용자 ID	varchar	사용자 ID
분야	varchar	관련분야
난이도	varchar	상 / 중 / 하
정답 수	int	
오답 수	int	
미응답 수	int	

표 3. 버전관리 테이블

필드명	데이터 형식	비고
파일명	varchar	yyyymmdd_itme.tsv
파일 용량	int	
생성 일자	varchar	파일 생성일
수정 일자	varchar	파일 수정일
반영 여부	char	Y / N
문제 수	int	문제 개수
반영 일자	varchar	unix_timestamp

3.3 문제 자동업데이트

문제 자동업데이트 기능은 TSV(tab-separated values)형태로 제작된 파일을 훈련서버로 가져와 데이터베이스에 추가하는 기능이다[그림3]. 훈련 서버는 crontab 스케줄러를 이용해 AUM¹⁾ 모듈의 SFTP Module을 호출하며 해당 모듈은 문제 제공 서버의 특정 위치의 파일리스트를 요청하고 파일 정보를 받아온다. 받아온 파일정보는 데이터베이스와 비교하게 되며 파일이 새로 추가된 경우 및 파일의 정보가 변경될 경우 해당 파일을 훈련서버의 특정위치로 가져온다. 가져온 파일은 TSV PM을 통해 문제 테이블[표 1]과 비교하여 입력 및 업데이트를 한다. 문제입력이 완료된 후에는 파일정보를 버전관리 테이블[표 3]에 저장 하고

프로세스를 종료하게 된다.

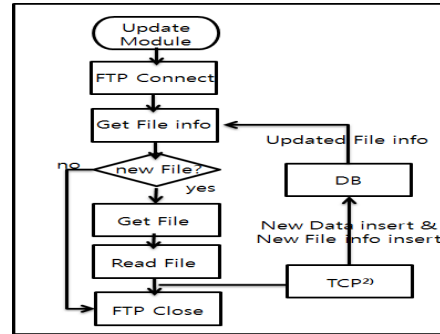


그림 3. AUM¹⁾ 시나리오

3.4 사용자 시나리오

사용자는 문제은행 사이트에 접속하여 문제를 선택하고 문제를 풀 수 있다. 만약 이전 통계데이터가 수집되어있다면 통계페이지를 통해 취약한 분야를 확인 후 문제를 풀 수 있다. 그리고 정답, 오답 결과는 페이지 상단의 그래프로 확인 할 수 있으며 서버부하를 줄이기 위해 페이지 자체에서 javascript로 채점한다. 또한 정답, 오답 결과는 사용자 ID를 기준으로 jQuery Ajax를 통해 훈련서버에게 전송된다[그림 4].

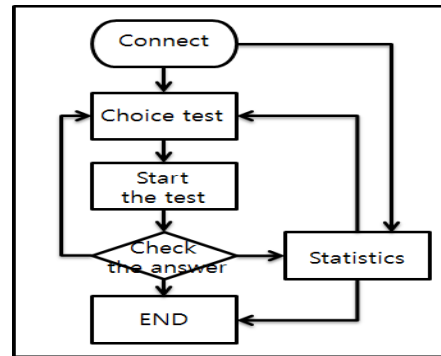


그림 4. 사용자 시나리오

IV. 구현

표 4. 개발 서버사양

OS	Ubuntu server 14.04.2 LTS
CPU	Intel(R) Core(TM) i5-4570
MEMORY	4G
Nodejs	v0.12.0

시스템 개발을 위한 서버사양은 [표 4]와 같다. 개발을 위해 최신버전의 Nodejs와 express를 설치했고, express 및 사용 모듈은 [그림 5] 와 같다.

```

{
  "name": "express4",
  "version": "0.0.0",
  "private": true,
  "scripts": {
    "start": "node ./bin/www"
  },
  "dependencies": {
    "body-parser": "~1.12.0",
    "cookie-parser": "~1.3.4",
    "debug": "~2.1.1",
    "express": "~4.12.2",
    "jade": "~1.9.2",
    "morgan": "~1.5.1",
    "serve-favicon": "~2.2.0",
    "mysql": "~2.6.1",
    "csv": "~0.4.1",
    "csv-parse": "~0.1.1",
    "stream-transform": "~0.0.7",
    "ssh2": "~0.4.6",
    "node-tsv-json": "~0.2.1"
  }
}
    
```

그림 5. packeage.json

모듈정보는 packeage.json 파일에서 관리되며, NPM(Node Packaged Modules)을 통해 버전관리와 모듈 설치 및 업데이트를 수행한다.

코드테이블은 정보보안 문제의 분야 및 난이도를 코드화 하여 저장했다. 이는 분야가 추가될 경우 시스템 수정 없이 추가 데이터만 입력하여 처리하게 했다. 데이터 베이스 논리구조는 [그림 6]과 같다.

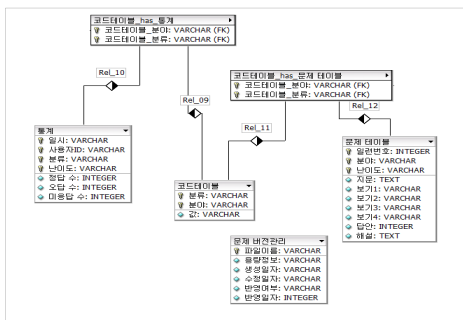


그림 6. 데이터 베이스 논리 구조

[그림 7]은 훈련 화면으로 답안제출을 클릭하면 정답, 오답, 미응답을 표시 된다. 그리고 백분율 하여 그래프로 표현한다. 통계 화면[그림 8]은 GoogleChart를 이용 했다. 사용자 훈련일시를 기준으로 일, 주, 월 통계를 지원하며, 사용자별 조화가 가능하다.



그림 7. 문제 화면(좌), 결과 화면(우)

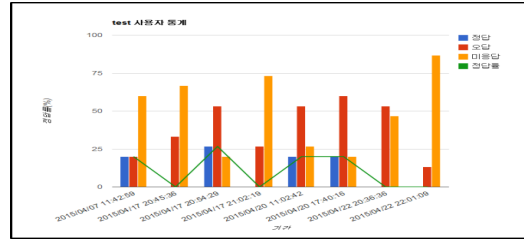


그림 8. 사용자 일별 통계

V. 결 론

약성코드 탐지건수에 비해 확연히 차이나는 신고 건수는 사용자의 보안의식이 크게 부족한 것을 알수 있다. 이러한 문제점들을 해결하기 위해 본 논문에서는 사용자 보안의식 강화를 위한 문제은행 방식의 교육훈련 시스템을 설계 및 구현했다. 안정적인 서비스를 위해 오픈소스 기반의 Apache 와 Nodejs를 사용하였고, 빠른응답 처리를 위해 jQuery Ajax및 json을 이용했다. 본 논문에서 구현한 시스템을 통해 사용자들의 보안의식 강화와 개인정보 침해사고를 예방할 수 있을 것으로 예상된다. 향후 연구로는 문제 제공을 빅데이터 데이터 마이닝 분석기술을 이용하여 자동 생성할 수 있는 연구가 필요하다.

This research was supported by the MSIP(Ministry of Science, ICT and Future Planning), Korea, under ICT/SW Creative Research program (NIPA-2014-H0502-14-3025) supervised by the NIPA(National IT Industry Promotion Agency)

참고문헌

- [1] 한국인터넷진흥원, ‘15년 1분기 스미싱 동향 분석 보고서, 2015. 04.
- [2] 한국인터넷진흥원, 2014년 10월 인터넷 침해사고대응통계, 2014. 10.
- [3] 전은아, 홍현술, 한성국, “효율적 평가체제 구축을 위한 XML기반의 문제은행 시스템 설계”, 한국인터넷정보학회, 2001 춘계학술발표대회 논문집 제2권 제1호, 2001.5, 133-138 (6 pages)
- [4] “시맨틱 웹 기반의 협업적 교육을 위한 문제은행 시스템”, 한국지능정보시스템학회 2006 춘계학술대회논문집, 2006.6, 270-276 (7 pages)
- [5] 김진식, 김송민, “NODE.JS + NGINX를 이용한 클라우드 기반 SNG(Social network Game) 제작을 위한 최적화 시스템 개발”, 한국정보기술학회 2013년도 하계종합학술·대학생논문 경진대회논문집, 2013.5, 128-130 (3 pages)
- [6] <http://expressjs.com/>