

Part 6. 애플리케이션 보안

Section 33. FTP보안

[0] 개요

- FTP(File Transfer Protocol) : 인터넷상의 컴퓨터들 간에 파일을 교환하기 위한 표준 프로토콜 (IETF RFC 959)
- 제어 연결(control connection) : Client → Server
 - 21번 포트 : 명령 또는 응답형태의 제어정보를 전송
- 데이터 연결(data connection) : 파일이 전송될 때 생성되는 데이터 연결
 - 20번 포트 또는 1024번 이후 포트 사용
 - 각 파일이 전송 될때마다 생성되며, 전송이 완료되면 폐쇄된다.

[1] 파일 관련 프로토콜

(1) FTP(File Transfer Protocol)

[FTP 연결]

- Default는 Active 모드이며, 모드 사용 여부는 클라이언트가 결정
- Active Mode : 서버에서 제어 포트 21번 ↔ 데이터 전송 20번 포트 사용 PORT 명령
 - 두 번째 connection은 서버에서 클라이언트로 접속
 - 문제점 : FTP 서버가 클라이언트 특정 포트에 접속하려는 시도가 외부 시스템 침입으로 간주되어 방화벽에 차단되는 경우가 있음
- 1) 클라이언트가 서버의 21번 포트에 연결하여 제어(control) 채널 생성
- 2) 서버가 21번 포트에서 클라이언트로 ACK 신호 보냄
- 3) 서버가 20번 포트(data port)에서 클라이언트가 알려준 포트에 연결 시도
- 4) 클라이언트가 서버의 데이터 포트(20번)로 ACK 신호 보냄
- Passive Mode : 서버에서 제어 포트 21번 ↔ 데이터 전송 1024번 이후 포트 사용 PASV 명령 사용
 - 두 번째 connection은 클라이언트에서 서버쪽으로 접속

(2) TFTP (Trivial File Transfer Protocol)

- 오직 기본적인 IP와 UDP만을 필요로 하는 FTP
- FTP는 TCP를 사용, TFTP는 UDP를 사용
- TFTP는 사용자 확인이나 비밀번호가 없으므로, 비밀번호 추측 공격은 무의미함

```
port 204.152.155.1.9.7
220 PORT command successful  -> PORT 명령 시 성공 응답
list
425 Can't build data connection : Connection refused
```

port 명령은 클라이언트가 임시 포트 번호를 선택하고 이를 수동 설정을 통해 서버에게 보낼 수 있는 명령이다.
220번은 FTP 상태코드로, 새 사용자에게 준비가 되었다는 뜻이고, 425번은 데이터 연결을 할 수 없다는 의미이다.
포트번호 2321 (=256*9+17을 요청하는 것이다


(3) NFS와 삼바

- NFS (Network File System) : TCP/IP 프로토콜 이용
- Samba :유닉스 계열과 윈도우 간 파일 및 프린터 자원 공유

[2] FTP 보안 위협 및 대책

[공격 유형]

- **FTP Bounce Attack**
 - FTP 서버가 데이터 전송시 목적지를 검사하지 않는 점 이용
 - 익명 FTP 서버를 이용해 그 FTP 서버를 경유해 호스트를 **스캔**

 -b : 바운스 스캔

- FTP PORT 명령을 사용
- FTP 서버를 통해 임의의 네트워크 접속을 릴레이함으로써 수행
- 다른 서비스가 20번 port 접속 요청시 거절하는 방법으로 해결
- 익명 FTP의 경우 임시적으로 ./incoming 파일 업로드를 제한

[보안 대책]

- SFTP (Secure File Transfer Protocol)

[3] FTP 서비스 운영

(1) proftpd

[특징]

- 매우 안정적이고 빠름
- xinetd / standalone 형태로 작동 가능

[접속 시 확인사항]

- /etc/passwd, /etc/shadow에 사용자 계정이 있는지 검사
- /etc/ftpusers에 사용자 id가 있으면 거부
- /etc/shell에 등록되지 않은 셸을 사용하는 유저 거부

(2) vsftpd (Very Secure FTP Daemon)

[주요 기능]

- 가상 IP별 별도의 환경 설정 기능
- 가상 사용자 지정
- 전송 대역폭 지정
- PAM 지원
- xferlog 표준 로그 파일보다 상세한 로그 파일 형식 지원
- Standalone 방식과 inetd(xinetd)를 통한 운영 모드 지원

Section 34. 이메일 보안

[1] 이메일 프로토콜

[0] 개요

- MUA (Mail User Agent)
 - 사용자 대행자. 메시지 작성, 읽기, 답장 보내기 등을 수행하는 S/W (아웃룩, 리눅스 pine)
- MTA (Mail Transfer Agent)
 - 메일 서버로 메시지를 전송하기 위한 전송 대행자. Client/Server (대표적으로 Sendmail, Exchange가 있음)
- MAA (Mail Access Agent)
 - 메시지를 검색하고자 할 때 사용하는 메시지 접근 대행자, Client/Server

[1] SMTP

- MTA, TCP 25번
- EHLO : SMTP 수신자에게 인사를 하면서 동시에 SMTP 확장 기능을 지원한다는 것을 알려준다.
- RCPT TO : 클라이언트가 메시지의 의도된 수신자가 누구인지를 알기 위해 사용
- MAIL FROM : 클라이언트가 전자우편의 송신자가 누구인지 알기 위해 사용

[2] MAA

- POP3 (Post Office Protocol ver 3)
 - TCP 110번
- IMAP4 (Internet Mail Access Protocol ver4)
 - TCP 143번
 - 추가 제공 기능 : 헤더 검사, 내용 검색, 부분 다운로드, 편지함 생성/삭제/이름변경

[2] 이메일 콘텐츠 보안 기술

[PEM] (Privacy Enhanced Mail)

- IEFT에서 채택한 기밀성, 인증, 무결성, 부인방지 제공
- 키 인증이 중앙집중화되어 있고, 이론 중심이어서 현재 많이 사용 X

[PGP] (Pretty Good Privacy)

- 구현이 용이, 소스 공개, 무료 사용
- 인증(전자서명), 기밀성(대칭 블록암호+RSA), 압축(ZIP), 호환성, 단편화 제공
- 송신 부인 방지 (O)
- 수신 부인 방지 (X)

[S/MIME] (Secure Multipurpose Internet Mail Extensions)

- MIME : 전자우편을 통해 ASCII가 아닌 데이터가 송신될 수 있도록 허용하는 부가적인 프로토콜
- 전자서명 : DSS 사용
- 세션키 암호화 : Diffie-Hellman 사용

- 전자서명 및 세션키 암호화 : RSA 알고리즘 사용

[3] 스팸 메일 보안 대응 기술

[메일 서버 등록제 SPF, Sender Policy Framework]

- 메일 헤더에 표시된 발송정보(IP)와 실제로 메일을 발송한 서버(IP)가 일치하는지 비교하여 **정보의 위변조 여부**를 파악할 수 있도록 하는 기술

[메일 서버 수신 차단]

- 콘텐츠 필터링, 송신자 필터링, 네트워크 레벨 필터링, 발송량 기준 차단, 시간대별 차단

[메일 서버 보안]

- 릴레이(relay) 스팸 방지, Anti-Spam 솔루션

[메일 클라이언트 보안]

- 콘텐츠 필터링, 송신자 필터링

[메일 서버 등록제 SPF (Sender Policy Framework)]

- 메일 헤더에 표시된 IP가 실제로 메일을 발송한 서버 IP와 일치하는지 비교함으로써 발송자 정보의 위변조 여부를 판단

[스팸메일 방지 보안 도구]

[SpamAssassin]

- 실시간 블랙리스트 분석 기법 사용
- 헤더 / 본문 내용 검사 (O)
- 첨부파일만 필터링 (X), MAC 주소 검사 (X)

[Sanitizer]

[Inflex]

[SpamAssassin]

[4] 기타 이메일 보안 대응 기술

[5] sendmail

- SMTP 프로토콜을 사용해 메일 서버 간에 메일을 주고 받는 역할을 한다.

Section 35. 웹 보안

[1] 웹 보안 개요

[2] WWW와 HTTP

[요청 메시지]

- 요청 라인(request line) : HTTP 요청 '메서드(get, put 등)
- 요청 헤더 라인(request header line) : 클라이언트의 웹 브라우저 종류, 쿠키, 콘텐츠 길이 등
- 본체(body)
- 메소드(method)

[응답 메시지]

[3] SSL/TLS (Secure Socket Layer, Transport Layer Security)

- 메시지 부인방지 서비스는 없다. (O)

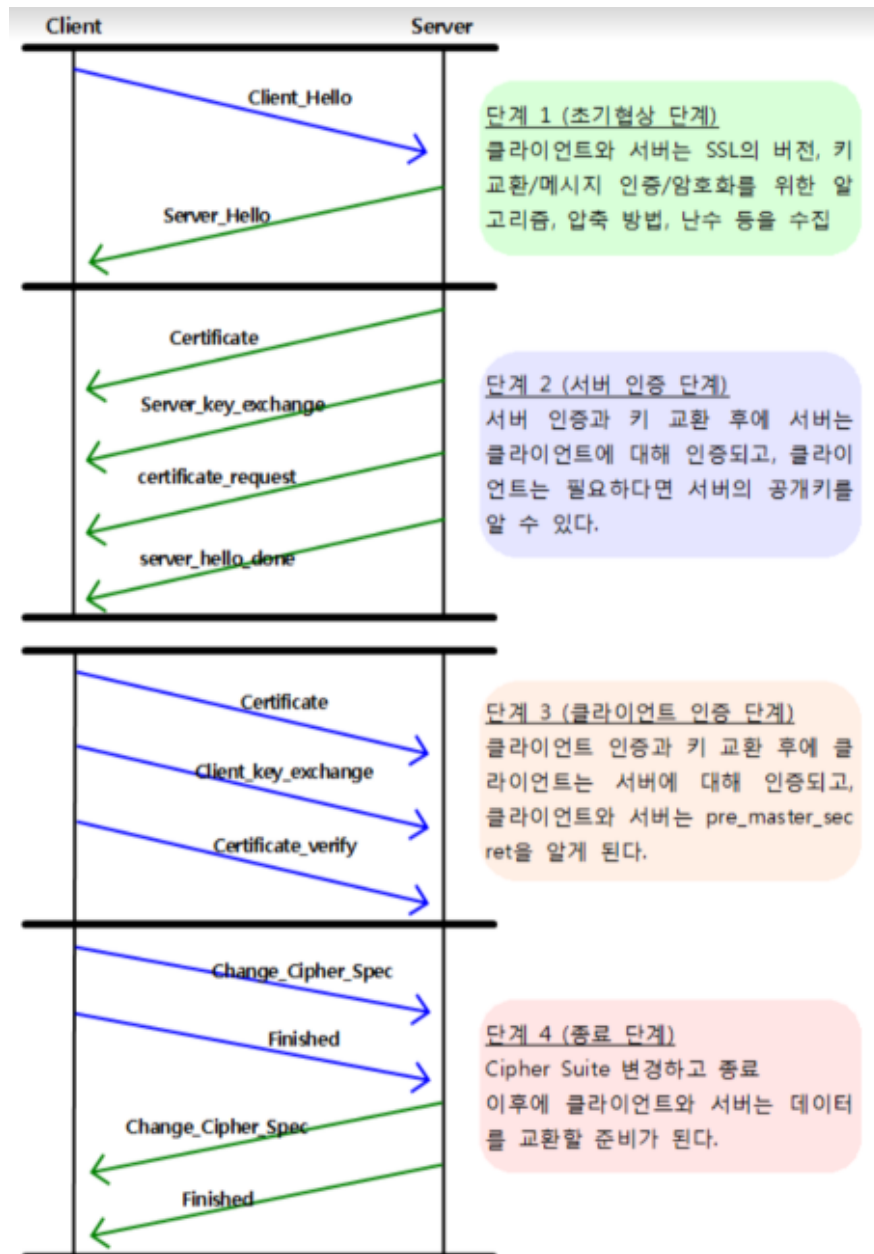
[SSL 보안서비스]

- 기밀성 : 대칭키 암호화 알고리즘 사용하여 제공하며, 비밀키는 Handshake Protocol을 통해 생성
 - 대칭키 암호화 알고리즘 : DES, RC4 등
- 무결성 : 해시 알고리즘을 사용하여 MAC를 만들어 메시지에 포함시켜 제공

[TLS 프로토콜]

- SSL 3.0을 기초로 IETF 표준

[Handshake 프로토콜] (#931)



- 암호화 알고리즘 **선택**
- 암호화 키 계산
- Record 프로토콜에 대한 보안 매개변수 제공
- 시도-응답 방식을 통한 서버와 클라이언트 간의 인증 수행

[Record 프로토콜]

- 대칭키를 이용하여 **메시지 캡슐화**
- 대칭키 암호화를 이용하여 메시지 캡슐화 수행
- 가장 하위에 위치한 프로토콜

[ChangeCipherSpec 프로토콜]

- 협상된 보안 파라미터를 이후부터 적용/변경함을 알리기 위해 사용

[Alert 프로토콜]

- 대등 개체에게 SSL 관련 경고를 할 때 사용됨.

[하트비트(Heartbeat) 프로토콜]

- 정상적으로 동작함을 알려주기 위해 생성하는 주기적인 신호

[4] 웹 서버 보안

[IIS 보안 설정]

[Apache 보안 설정]

[5] 웹 보안위협 및 대응책

[6] 소프트웨어 개발 보안

Section 36. DNS 보안

[1] 호스트 설정과 호스트 설정 프로토콜

[2] DNS

[3] DNS의 보안

[4] DNS 서버 보안 설정

DNSSEC(Domain Name System Security Extensions)는 인터넷 프로토콜(IP) 네트워크에 사용되어 도메인 네임 시스템(DNS)이 제공하는 특정한 종류의 정보를 취득하기 위한 국제 인터넷 표준화 기구(IETF) 사양을 위한 스위트의 하나이다. DNS 클라이언트에게 DNS 데이터의 메시지 인증, 실체에 대한 인증 거부, 데이터 무결성을 제공하지만 가용성이나 비밀 보장을 제공하지는 않는 DNS에 대한 확장 집합이다.

Section 37. 데이터베이스 보안

[1] 데이터 베이스 기본 개념

[2] DB 보안 요구사항

[3] DB 보안 통제

[4] 클라우드 보안

[5] DBMS 보안 관리

Section 38. 전자상거래 보안

[1] 전자상거래 개요

[전자상거래 개념]

[전자지불 시스템의 종류]

- 네트워크형 프로토콜 : 인터넷과 같은 네트워크 환경에서 사용자의 PC나 서버의 계좌 등에 자신의 전자화폐를 저장 및 사용하는 프로토콜
 - Millicent, NetBill, Payword, E-cash, 국내(Ecoin, iCash, EasyCash)
- 가치저장형 프로토콜 : 스마트카드 내에 전자화폐를 저장하고 사용하며, 인터넷과 같은 네트워크 환경보다는 실생활의 화폐를 대응키 위한 목적으로 구성된 프로토콜
 - Mondex, VisaCash, Proton, ChipKnip, 국내(K-Cash)

[전자화폐]

- 요구조건
 - 재사용 불가능성(=안정성, security)
 - 익명성(Privacy, Un-traceability, 추적불가능성)
 - 오프라인성(Offline) : 지불 단계를 오프라인으로 처리할 수 있어야 함
 - 양도성(Transferability)
 - 분할이용 가능성
 - 부정사용자의 익명성 취소
 - 이중사용 방지

[전자지불 시스템]

- 보안요구 사항
 - 위조불가능(Unforgeability)

- 부인방지(Non-Repudiation)
- 누명면제(Framing-Freeness)
- 무결성 인증
- 비밀성(Privacy) / 익명성(Anonymity)

[2] 전자상거래 정보 보호

[3] SET(Secure Electronic Transaction)

[SET 프로토콜 개요]

- Visa와 MasterCard 합동 개발
- RSA 암호기술 사용
- 전자결제 시 교환되는 정보의 비밀 보장을 위해 공개키, 비밀키 암호 알고리즘을 사용한다.
- 데이터 무결성을 확보하고자 전자서명과 해시 알고리즘을 사용한다.
- 주문 정보는 상점의 공개키로 암호화하고, 지불 정보는 은행의 공개키로 암호화한다.
- 지불 정보와 주문 정보는 상점과 은행이 상호 협조하여 모두 볼 수 있도록 구성되어 있다.
(X)
→ 고객의 지불정보(신용카드 번호 등)은 상점이 알지 못하게 하고, 주문 정보는 은행이 알지 못하게 함으로써 고객의 프라이버시를 보호한다.
- RSA 동작은 프로토콜의 속도를 크게 저하시킨다.
- 카드 소지자에게 전자지갑 소프트웨어를 요구한다.
- 지불게이트웨이에 거래를 전자적으로 처리하기 위한 별도의 하드웨어와 소프트웨어를 요구한다. (O)
- 고객은 개인키와 공개키를 모두 가진다. (O)

[SET 프로토콜의 목적]

- SSL의 단점을 극복한다 (O)
→ 상인에게 지불정보 노출을 해결한다,
- 암호 프로토콜의 단순화 (X)
→ 복잡하다는 단점 가지고 있음
- 정보의 기밀성 확보
- 지불 정보의 무결성 확보

- 상인과 고객의 사옹 확인

[SET 참여주체]

- 카드소지자, Cardholder(Customer)
- 발행자, Issuer (Cardholder의 은행)
- 가맹점, Merchant
- 지불은행, 매입사, Acquirer : 판매자의 가맹점 승인 금융기관 혹은 판매자의 계좌가 개설되어 있는 금융기관
- 지불 게이트웨이(Payment Gateway) : 지불처리 은행 또는 제3자에 의해 운영되는 시스템
- 인증기관(CA, Certification Authority) : SET 참여자에게 공개키 인증서를 발행하는 기관

[이중서명 프로토콜]

- 카드사용자가 구매정보와 지불정보를 각각 해시한 후, 두 해시값을 합한 뒤 다시 해시한 값을 카드 사용자의 개인키로 암호화(서명)하여 이중서명값 생성
(두 해시값은 연관되어있어야 함)

[4] 전자상거래 응용보안

[ebXML]

- 인터넷 표준 브라우저만으로 장소에 구애 없이 어디서나 전자상거래 할 수 있음, 구현 비용 저렴

[ebXML 구성요소]

- 비즈니스 프로세스
- 핵심 컴포넌트
- 등록 저장소
- 거래 당사자
- 전송, 교환 및 패키징

[무선플랫폼에서의 전자상거래 보안]

- WPKI(Wireless Public Key Infrastructure)

[5] 기타

[지급결제서비스]

- 전자지급결제대행(PG) : 구매자로부터 대금을 수취하여 가맹점(판매자)에게 최종적으로 지급될 수 있도록 결제정보 전송과 정산업무를 대행
- 결제대금예치(Escrow) : 공신력있는 제3자가 구매의 결제대금을 예치받고, 구매자에게 무효가 전달되었는지를 확인한 후 대금을 가맹점에 지급하는 서비스

Section 39. 각종 애플리케이션 보안위협 및 대책

BYOD(Bring Your Own Device) 보안 기술

- MDM(Mobile Device Management)
- 컨테이너화
- 모바일 가상화(Hypervisors)
- MAM(Mobile Application Management)
- NAC(Network Access Control)