1.



정보보호의 기본원칙 3가지(CIA)

**1. Confidentiality(**기밀성)

: 허락 받지 않은 적이 데이터를 가져가지 못하도록 하는 것(Data confidentiality, Privacy)

: **is the avoidance of the unauthorized disclosure of information**

🡺 **Tools for confidentiality**

1. Encryption: transformation of information using an encryption key

2. Access control: Rules and policies that limit access to confidential information to people with “need to know”

🡺 알 필요가 있는 사람한테 접근 권한

🡺 Serial number, role로 판단한다

3. Authentication: determine the identity

🡺 something the person has (카드, 키)

🡺 something the person knows (패스워드)

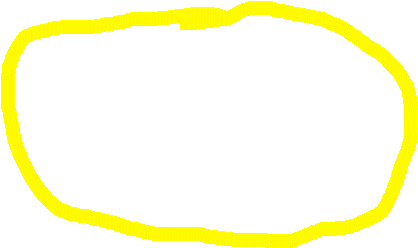
🡺 something the person is (지문, 홍채)

4. Authorization: if person is allowed access, base on an access control policy

5. Physical security: 물리적인 제한을 두는 것

🡺 locks on cabinets and doors

🡺 place computers in windowless rooms



**2. Integrity**(무결성)

: 허가 받지 않은 사용자가 시스템을 마음대로 수정하지 않도록 하는 것(Data integrity, System integrity)

: **The property has not be altered in an unauthorized way** (무단으로 변경 못 하도록 함)

🡺 **Tools for integrity**

1. Backup: The periodic archiving of data

🡺 일정기간 데이터 백업

2. Checksum: computation of a function that maps the contents of a file to a numerical value

3. Data correcting code: easily detected (쉽게 변화 감지)

**3. Availability**(가용성)

: 권한이 있는 사람이 자기가 원하는 시간에 정보에 접근할 수 있도록 하는 것

: **is accessible and modifiable in a timely fashion by those authorized to do so**

🡺 **Tools for Availability**

1. Physical protections: infrastructure meant to keep information available (물리적 공격에 방어)

2. Computational redundancies: data will be copied to other computer and storage devices

🡺 데이터 이중화

**4. Authenticity**: 얼마나 정확한 지 유지시키는 것

**5. Accountability**

: keep records for tracing when the problem occurred (문제 발생 시 추적을 위해 기록 저장)

컴퓨터 보안 용어

1. Adversary: an entity that attacks (공격하는 개체, 적)

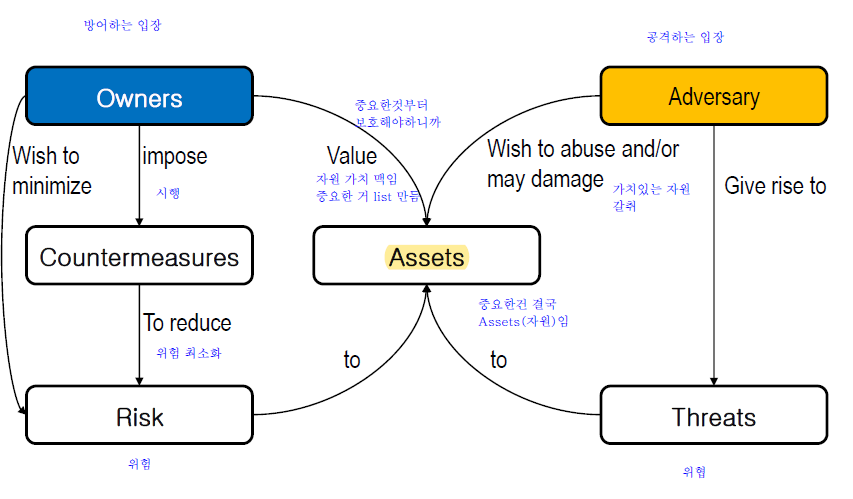
2. Attack: Deliberate attempts to evade security service and violate the security policy of a system (보안 정책 위배)

3. Assess: HW, SW, data, network 등

4. Countermeasure: action, device that reduce a threat

🡺 threat을 줄이는 기술





Vulnerability(취약점의 종류)

1. corrupted (loss of integrity)

2. leaky (loss of confidentiality)

3. unavailable or very slow (loss of availability)

Attack(공격의 종류: threats carried out)

1. Passive: does not affect system resources

2. active: attempt to alter system resource

3. insider: initiated by an entity inside the security perimeter

4. outsider: initiated from outside the security perimeter

위협과 공격 유형

Eavesdropping(도청)

: The interception of information intended for someone else during its transmission over a communication channel. (통신하는 동안 가로채기)

Denial-of-service(DoS)

: The interruption or degradation of a data service or information access (서비스 중단되거나 느리게 만듦)

Alteration

: unauthorized modification of information (권한 없이 수정)

Masquerading

: 실제 사람이 아닌데 그 사람인 척하고 보내는 것

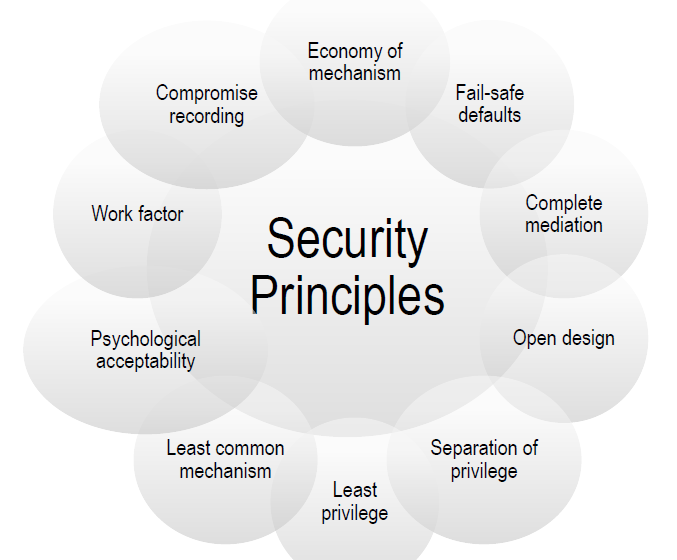
Repudiation

: denial of a commitment or data receipt (데이터 거절)

Correlation and traceback

: determine the source of a particular data stream (추적)

보안원칙 열 가지



1. economy of mechanism: simple security framework can be understanded by developers (단순하게)

2. Fail-safe defaults: do not favor usability over security 🡺 편의성에 너무 치우치지 말고 보안성도 생각해라

3. Complete mediation: protection scheme 따르자

4. Open design: security architecture이나 design은 공개될 수 있고 구조를 재사용하자

5. Separation of privilege: 자원 접근 시 여러 조건 둬야 함

6. least privilege: 기능 다 알고 있어야 함

7. least common mechanism: 여러 사용자 사용 시 주의

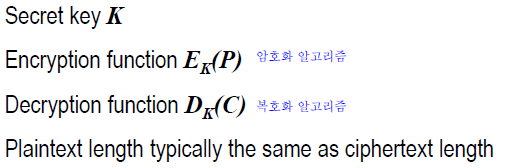
8. Psychological acceptability: 전문가 외에도 세팅 가능

9. Work factor

10. Compromise recording: 문제 발생시 기록하자

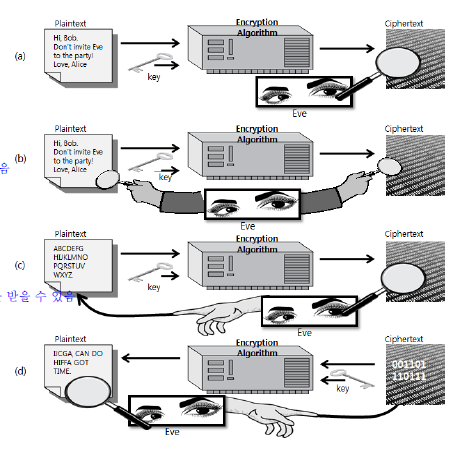
**2.**

Symmetric cryptosystem



여기서 P와 C는 **P**laintext: 평문, **C**iphertext: 암호문





a : collection of ciphertext 🡺 방어하기 쉬움

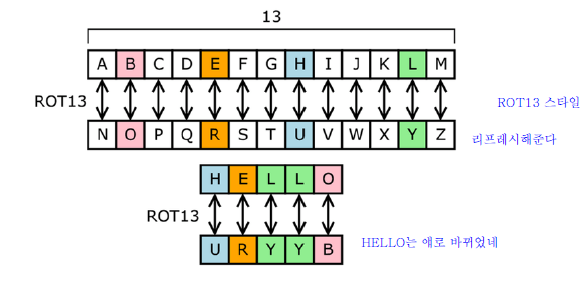
b : collection of plaintext/ciphertext pairs

c : collection of plaintext/ciphertext pairs for plaintexts selected by the attacker by the attacker

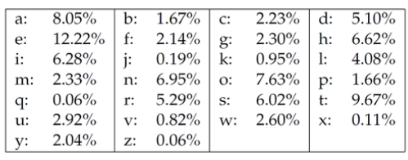
Brute-Force attack

: Try all possible keys K and determine if DK(C) is a likely plaintext 🡺 시도할 수 있는 키 다 해보는 것

Substitution Ciphers

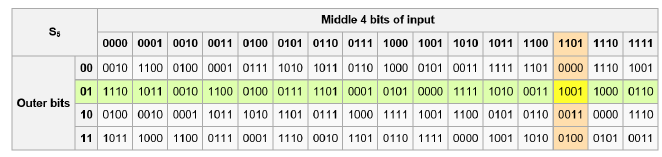


* Each letter is uniquely replaced by another
* **Frequency Analysis**로 뚫릴 수 있음

: knowledge of letter frequencies can be used in attack

**Substitution boxes**

: substitution in binary number



One-time Pads

: one type of substitution cipher that is absolutely unbreakable

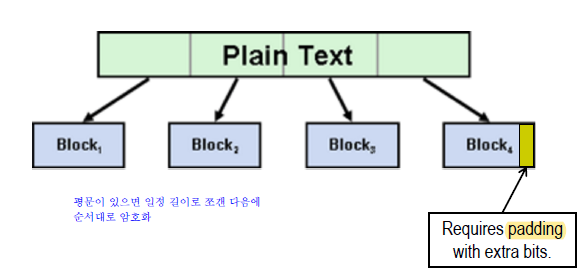
* Use a block of secret keys to encrypt a plaintext M, or length n, with each secret key being chosen uniformly at random
* 단점 : the key has to be as long as the plaintext (키 길이가 평문 길이만큼 되어야 함)

🡺 keys can never be reused(키 다시 사용 불가)

**Block ciphers**

: each message is divided into a sequence of blocks and encrypted of decrypted in terms of its blocks

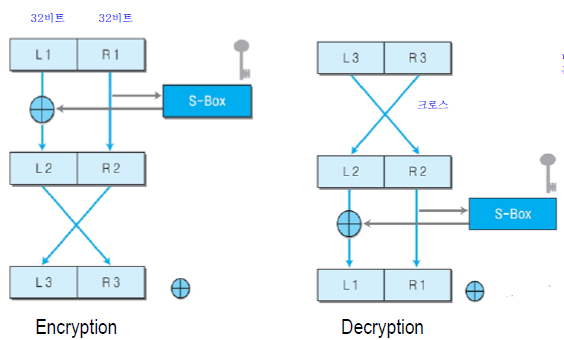
* 각 메시지는 block으로 쪼개고 암호화 복호화



* ****마지막 블록은 일정한 길이 아니기에 padding이 요구된다

대표적인 블록 알고리즘, Data Encryption standard**(DES)**

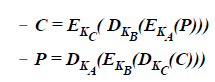
: **64bit block and 56bit keys**



**Triple DES(3 DES)**

: use three different keys KA, KB, Kc

키 세 개 쓰니까 56 \* 3 = **168bit key length**

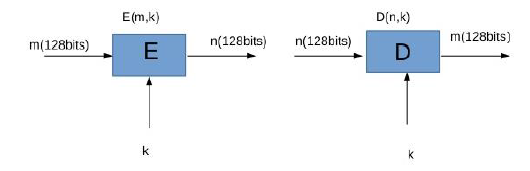




* 근데 DES는 키 사이즈가 여전히 너무 작아

AES: **block 128 bit**, **key 128, 192, 256 bit**

AES-128 / AES-192 / AES-256



**3**

**Private-key cryptography**

: Traditional private key cryptography use one key

: shared by both sender and receiver(모두 공유)

: if key is disclosed, communications are broken

: also is **symmetric**

Public-key cryptography

: use two keys – public & private

: **asymmetric**

: use clever application of **number theoretic**

* 수학적 개념이 들어감

: complements rather than replaces private key crypto

Why use public-key cryptography?

1. key distribution: how to have secure communications

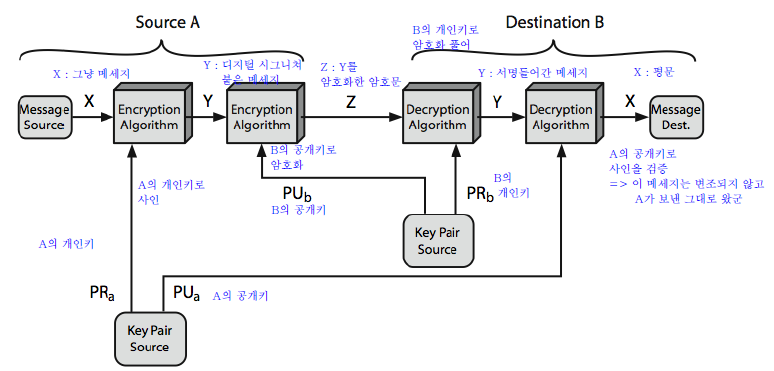
2. digital signatures : how to verify a message comes intact from the claimed sender(디지털 서명 어떻게 앎?)

Public key used to encrypt messages, verify signatures

* Known by anybody(누구나 앎)

Private key used to decrypt messages, and sign signatures 🡺 known only to the recipient

* 키 두개이기에 asymmetric
* 공개(암호화)키만 알면 infeasible
* 두 키 다 알면 해석 쉬움



공개키(public key)로는 암호화, 서명 검증

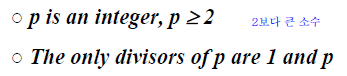
개인키(private key)로는 복호화, 서명

Public key schemes는 brute force 가능하지만 큰 수는 엄청 오래 걸림 🡺 그래서 **key는 512비트 이상**

일반적인 Private algorithm**(RSA algorithm)**

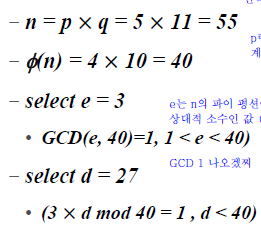
* Using one-way function







RSA 방식



1. 엄청 큰 소수 p, q를 곱해서 n을 구한다

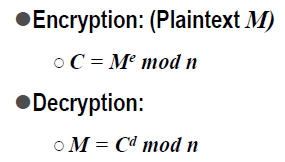
2.  (p-1\*q-1) 하면 나옴 (약수 개수)

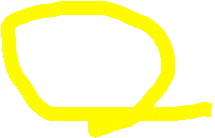
3. 1보다 크고 보다 작은 서로소(e) 구해

4. e \* d mod  = 1 이 되는 d 값을 구해

🡺 e \* d / 하면 나머지가 1이 나오는 값

5. 그럼 public key(e, n), private key(d, n)

****



**Diffie-Hellman key agreement**

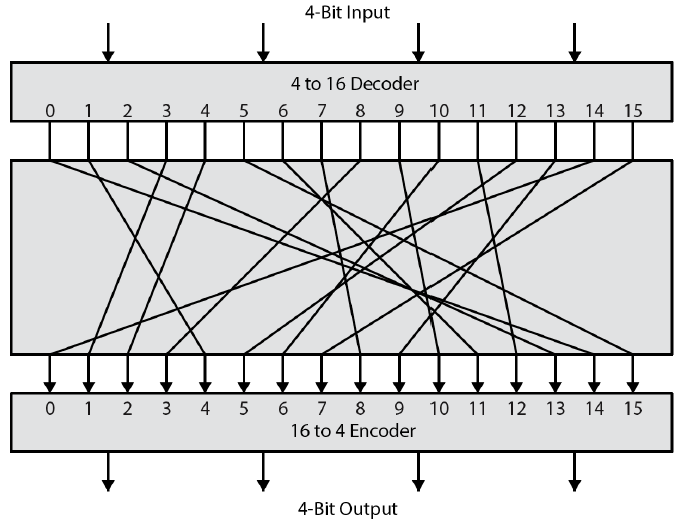
🡺공개키 알고리즘을 이용해서 대칭키를 나눠주는 방식

**4**

Block cipher: 메시지가 block에 저장되어서 암,복호화

Stream ciphers: process messages a bit or byte at a tie when en/decrypting

* **Feistel Cipher structure 기반으로 함**
* 64bit 블록 처리 시 264 테이블 필요



Substitution-Permutation networks

1. substitution(S-box)

2. permutation(P-box)

🡺 대칭 키 알고리즘은 이 두개 과정의 반복이다

🡺 provide confusion & diffusion of message & key

Cipher needs to completely obscure statistical properties of original message 🡺 암호는 통계학적 속성을 완전히 없애야 한다

* **One-time pad** do this

하지만 더 실용적으로 S&P를 통해 얻을 것을 제안

**Diffusion**: remove statistical structure of plaintext over bulk of ciphertext 🡺 통계 구조를 소멸시킨다

**Confusion**: makes relationship between ciphertext and key as complex as possible 🡺 혼동을 통해 키와 암호문 관계가 복잡해짐

**Feistel Cipher Structure**

1. partitions input block into two halves

🡺 여러 round를 통해 process

🡺 left data half: substitution

2. implements shannon’s S-P net concept

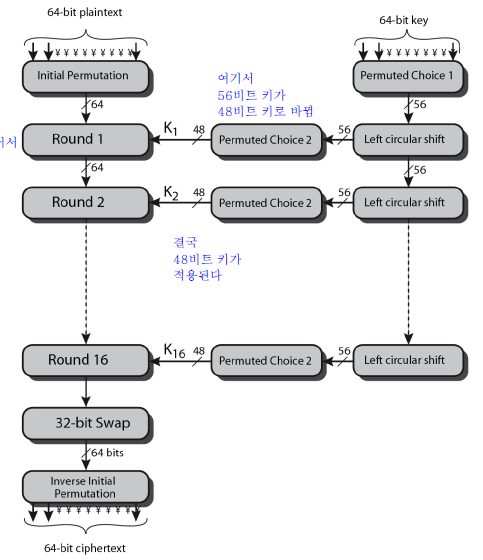
대충 이런 모양

**feistel cipher design elements**

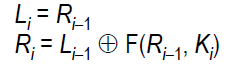
: block size, key size, number of rounds, round function,

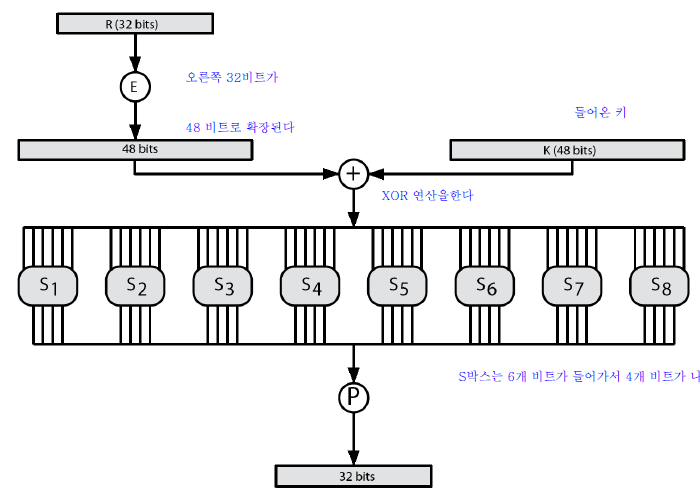
fast SW en/decryption, ease of analysis

DES도 feistel cipher structure 사용해

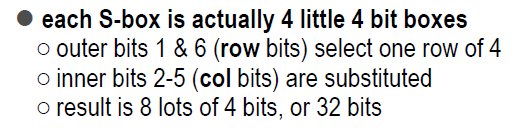


* 32비트씩 쪼개진다





* 32비트 데이터 들어와서 48비트로 확장돼
* 48비트 키랑 XOR 연산 후 32비트로 나와
* 이때 사용되는 것이 S-Box



* 가장자리 두개가 4개 중 1개 선택
* 안에 4개로 컬럼 선택

DES Key

키는 기본적으로 56 bit 선택하는데 28비트로 쪼갠다

16번 스테이지를 통해 이루어져 있다

복호화는 역순이다

* Unwind steps of data computation

 🡺subkey



**Hash function**

Hash function h에 평문 P 넣으면 x값이 나온다

***x = h(P)* called hash value or digest of *P***

만약 ***h(P) = h(Q)***이면 collision 발생한 것

* Collision은 피할 수 없다
* One way 기능을 가지고 있어야 함

(h(P)=x는 쉽지만, x를 P로 바꾸는 건 어렵)

* Weak collision resistance 대비해야 해

(같은 해시밸류 가져도 평문 찾기 어려워야 함)

* Strong collision resistance

(collision 찾기 어려워야 해)

* Hash value는 256비트 정돈 해야 해

Iterated hash function(반복 해시 함수)

* MD5 and SHA



**5.**



**Authentication**: determination of identity

🡺process of verifying that identity

* 아이덴티티를 결정하는 것

4가지 방식

**Something the person has(smart card, radio key)**

**Something the person knows(password, PIN)**

**Something the person is(fingerprint, face)**

**Something the person does(handwriting, voice)**

Fundamental building block and primary line of defense 🡺 제어의 가장 기초가 되고 있음

두 개의 step으로 나뉨

1. identification step

🡺 presenting an identifier to the security system

🡺 인증정보를 주는거야, 아이디 패스워드 제출

2. verification step

🡺 generating authentication information that corroborates the binding between the entity and the identifier 🡺 인증 정보 생성

* Verifying 과정, 너가 맞는 지

Authentication 후에 authorization을 한다

**Authorization**: an anonymous agent could be authorized to a limited action set 🡺 접근 권한을 줌

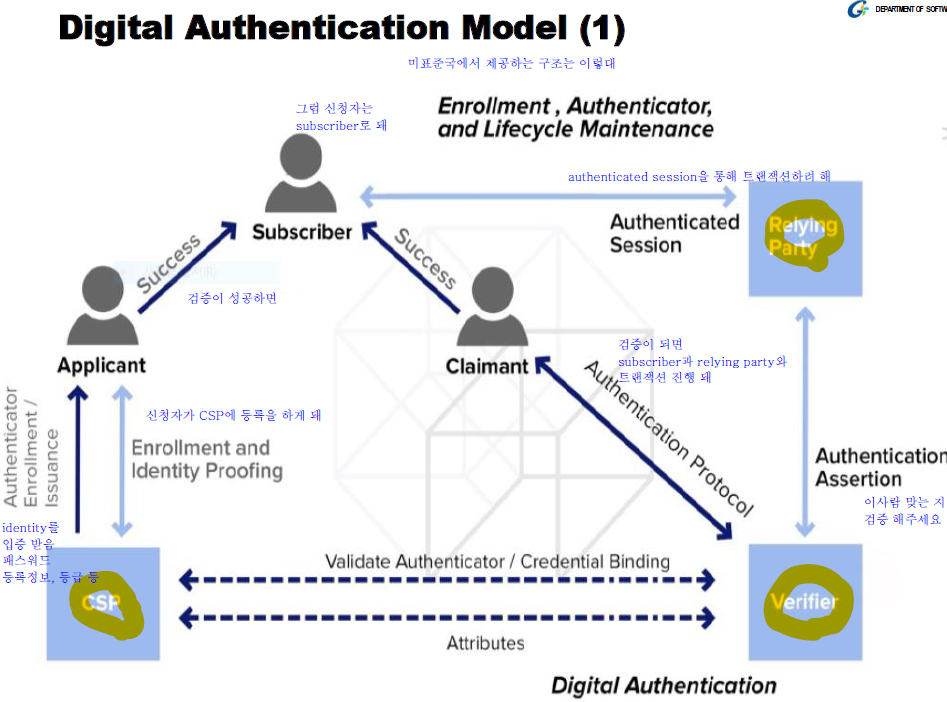
Authentication protocol

* 서로 확인하고 세션 키 교환
* One way or mutual

Single-factor authentication : the weakest level of authentication 🡺 인증 중에 가장 약한 애들

* 위에 방식 중 하나만 사용하는 것

Multi-factor authentication : use of more than one of the authentication means 🡺 2개 이상 방식 사용 함



디지털 환경 authentication model

* Enrollment 🡺 처음에 신청자가 CSP에 등록을 하게 되며 이제 subscriber가 된다
* Authentication: subscriber는 token이나 user information을 제출하고 허가를 받게 되면 online transaction을 하게 된다
* Life-cycle maintenance: CSP는 credential 잘 유지해야 하고, subscriber은 그들의 authenticator을 잘 유지할 책임이 있음

**What do you know**

* User provides name/login and password
* User id determine user’s authorization to access the system

: how is password stored?

* Hash function h : strings 🡺 strings
* Password 입력 시 h function 계산해서 저장되어 있는 파일과 비교한다, disk엔 저장되지 않음

🡺offline dictionary attack, specific account attack, popular password attack 등 발생 가능

Password selection techniques

1. user education

2. computer generated passwords

3. reactive password checking

**What you have**

Digital authentication : physical devices to aid authentication 🡺 인증을 도와주기 위한 물리적인 디바이스

Ex : eToken, smart card, RFID

**eToken** : can be implemented on a USB key or a smart card

* Data physically protected on the device itself
* One-time password를 위해서 사용된다
* Storing password, digital signatures, private key



종류 : memory card, smart token

**Memory cards**

* Can store but do not process data(연산X)
* Magnetic stripe card
* Can be used alone for physical access(hotel)

(물리적으로 단독으로 쓰일 수 있음)

단점 : requiring special reader(리더기가 필요해)

Loss of token(token 잃어버릴 수 있음)

**Smart token**

* Including an embedded microprocessor

(프로세서를 포함하고 있음, 연산 가능)

* Smart token 형태
* Display가 포함되어 interface 볼 수 있음
* Read/writer 통신하는 electronic interface 있

**Who you are**

: biometric authentication

Biometric reading

**Physical biometrics**

* Fingerprint, iris, retina, hand geometry

Finger geometry, face geometry

**Behavioral biometrics(dynamic biometrics)**

* Signature, voice, keystroke, gait

**6**

**Malware**: SW intentionally designed to cause damage to a computer, server, client or computer network 🡺 의도적으로 컴퓨터, 서버, 클라이언트 네트워크 손상시키기 위한 SW

* 종류 : worm, computer virus, trojan horse, ransomware, spyware and scareware

Spyware: programs designed to monitor user’s web browsing, display unwanted advertisement 🡺 원치 않는 광고 보여주고, 유저 웹 브라우저 모니터링

Ransomware: affects an infected computer system in some way, and demand payment to bring it back to its normal state

* 감염시키고, 돈을 요구함

🡺Malware 중에 money-making이 목적이 아니라 sabotage가 목적인 애들도 있다

**Viruses and worms**



Computer virus is SW that embed itself in some other executable SW 🡺 실행 가능한 SW에 embed한다

* When SW run, the virus is spread to other executables

Worm is a stand-alone malware that actively transmits itself over a network to infect other computers

Concealment

Virus: 복사본 만들어서 숨는다

Trojan horse: 정상적인 프로그램인 척 숨음

Rootkits: 호스트의 OS를 수정함

Backdoors: 사용자에게 보이지 않게 액세스 할 수 있는 것, 인증 절차 우회할 수 있는 것

Anti-malware strategies

1. real time protection

2. detection and removal

Grayware

Unwanted application or files that are not classified as malware and may cause security risks

* Spyware, adware가 여기에 들어 감

Virus: 자신의 복사본을 반복적으로 복제하는 바이러스 코드입니다.

Worm: 네트워크에서 스스로 복제되는 바이러스

* 막기 위해선 어떤 파일인지, 포맷인지 어느 환경에서 작동되는 지 파악해야 함

**7**

Insider threat가 발생하는 세가지 이유

1. 내부에서 일어날 거라고 주의를 하지 않기에

2. 솔루션이 개인의 audit data source에 의존

3. 규칙을 구축하는 데에 도메인 지식에 의존함

Insider은 크게 세가지로 나눔

Traitor: action are counter to policy

* Legitimate(합법적인) 사용자임, 악의적인 마음을 가지고 있음
* Malicious(악의적인) 행동을 한다

Masquerader: attacker who succeeds in stealing a legitimate user’s identity

* 사용자의 identity를 훔쳐서 악의적인 목적을 위해 정상적인 사용자처럼 가장하는 것,

Unintentional perpetrator: a legitimate user unintentionally making a mistake 🡺 합법적인 사용자인데 실수함

Insider threats(내부자 위협)

1. data exfiltration (데이터 유출)

2. violations against data integrity (중요한 데이터 수정)

3. sabotage of ICT system

**Masquerader 🡺 kill chain**

1. Reconnaissance: scan 하는 과정

2. weaponization: social engineering

3. delivery: email spam

4. exploitation: privilege escalation

5. installation: backdoor

6. command and control(C2): DDoS

7. actions on objectives: Data exfiltration

**Traitor, unintentional perpetrator 🡺 actions on objectives 바로 간다**

**Host-based analytics**

🡺OS low-level data 부터 application level data까지 다 모은다

🡺system call을 통해 분석한다

🡺traffic을 모아서 분석한다

🡺host log를 분석한다

분석할 데이터

1. host-based data sources(호스트 기반 데이터)

System call : is the channel a program communicates with the kernel of the OS

Command, keyboard, mouse를 통해 분석



이런 것들

Host log provided rich data source for tracing a host’s behavior

* Host log에는 이벤트 기록이 남아있음, 추적하기 좋은 풍부한 데이터를 제공한다.

2. network traffic and logs(네트워크 기반)

Network traffic 정보를 모은다

Network log는 characterizing how users of hosts behave in a network이다 🡺 네트워크에서 유저가 어떻게 행동했는 지 보여주는 것

네트워크 로그 분석 한다.

DHCP, 이메일 등에서 수집된다

3. contextual data sources(심리 의도)

**Backdoor**

: hidden feature or command in a program that allows a user to perform actions he or she would not normally be allowed to do 🡺 프로그램 내에서 하라고 허락 받지 않아도 수행할 수 있도록 하는 명령어

**Logic Bomb**

A logic bomb is malware that is triggered by a response to an event, such as launching an application or when a specific date/time is reached 🡺 특정 날짜, 시간이 되었을 때 같은 이벤트에 대한 응답으로 트리거 되는 malware

**8**

**Virus**: computer code that can replicate itself by modifying other files or programs to insert code. 🡺 코드에다가 자신을 심어 수정을 통해 복제

Virus phases(바이러스 단계)

1. Dormant phase

🡺 조용히 있는 단계

2. Propagation phase

🡺 infecting new files on new system

3. Triggering phase

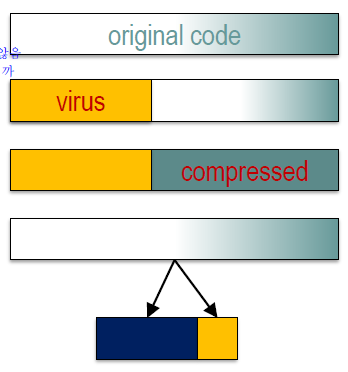
🡺 logical condition causes to perform

(특정 시기가 되면 작동)

4. Action phase

🡺 액션 하게 된다

Virus infection types(감염 타입)



Overwriting

* Destroys original code
* 기존 것 덮어씌우는 것

Pre-pending

* Keeps original code, possibly compressed
* 원래 코드 압축, 그 앞에 붙임

Marco viruses

* Infects MS office documents

Virus concealment



1. Encrypted virus

🡺 decryption engine + encrypted body

🡺 randomly generate encryption key

🡺 detection looks for decryption engine

:: encryption key가 계속 바뀌는 애

2. polymorphic virus

🡺 encrypted virus with random variation of the decryption engine

🡺 detection using CPU emulator

:: decryption engine이 계속 바뀌는 애

3. metamorphic virus

🡺 different virus bodies

Virus body가 계속 바뀜

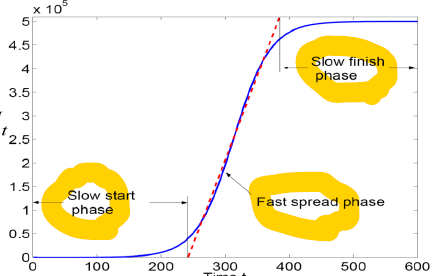
* Hard to detect

**Worm**: malware program that spreads copies of itself without the need to inject itself in other program

Forward edges: infection

Back edges: already infected or not vulnerable

**Worm propagation**:



1. Slow start phase

2. fast spread phase

3. slow finish phase

**Trojan horse**

: malware program that appears to perform some useful task, but which also does something with negative consequences 🡺 도움되는 프로그램처럼 보이지만 부정적인 결과를 초래하는 프로그램

**Rootkits**

Set of hidden programs installed on a system to maintain covert access to that system 🡺 시스템에 은밀하게 접근해서 installed된 hidden program 종류

**Malware Zombies**

Malware can turn a computer in to a zombie, which is a machine that is controlled externally to perform malicious attacks, usually as a part of a botnet

1. bot SW infect some computer

2. botnet controller connect with zombie

3. botnet controller command zombie

4. when connect process done, action to victim with botnet

**Adware**: attack for getting user’s privacy and sensitive information 🡺 display unwanted advertise

**Spyware**:collect sensitive data at background and send to spyware data collection

**A malware countermeasure**

1. signatures: scan compare the analyzed object with a database of signatures

🡺 signature is a virus fingerprint

**Signatures database**

CME, DIS가 있다

**White list**

: SW which is considered safe to run, blocking all others

**Black list**

: unsafe SW, item on the list are denied access

Suspicious file can be isolated in a folder called quarantine

* 의심스러운 파일은 quarantine 폴더에 고립될 수 있음
* 결과를 signature에 업데이트



시험문제 8.

UNIX password schema가 무엇인가?

시험문제 9.

Known plaintext attack이 무엇인가?