**1. Introduction**

Threats A loss or harm that might befall a system 1 Interception 2 Interruption 3 Modification 4 Fabrication

Methods of defence

• Prevent it: • Deter it: • Deflect it • Detect it: • Recover from it • “Defence in depth”

Security Attacks

• A passive attack The release of message contents – Traffic analysis

• An active attack Masquerade Replay Modification of messages Denial of service

The Basic Components

1) • Confidentiality: prevent unauthorized reading of information – Cryptography used for confidentiality

2) • Integrity: detect unauthorized writing of information – Cryptography used for integrity

3) • Availability: Data is available in a timely manner when needed, Denial of service (DoS) attacks

**2. Classical Encryption Techniques**

• Only the key is secret crypto algorithms are not secret • This is known as Kerckhoffs’ Principle

• Basic types of Symmetric Encryption (Substitution, Permutation, Transposition)

1) Simple Substitution “Caesar’s cipher”

• A simple substitution (shift by n) is used • Only 26 possible keys! • Exhaustive key search **“Cipher Wheel”**

2) Permutation

Then 26! > 288 possible keys! • English letter frequency counts…

3) Vigenère Substitution

Mapping table, repeat key many times, key is less than or equal with message

4) Double Transposition

Encryption Operations: permute rows Then transpose the columns

❑ Key is matrix size and permutations:

5) One-Time Pad Vernam cipher

– Ciphertext provides no info about plaintext – All plaintexts are equally likely– Pad must be random, used only once – Pad is known only to sender and receiver • Note: pad (key) is same size as message

Claude Shannon

– Confusion ⎯ obscure relationship between plaintext and ciphertext ciphertext 분석해도 plaintext 모름, vegenere crypto system은 ciphertext letter frequency 로 문제 풀 수 있어서 no, substitution

– Diffusion ⎯ spread plaintext statistics through the ciphertext ciphertext frequancy분석해도 plaintext 모름, ciphertext의 반복은 uniform 해야 함, permutation

• One-time pad is confusion + diffusion(fully random key), while double transposition is diffusion-only

**3. Symmetric Encryption**

• Block ciphers operate on a block of data – entire block must be available before processing

• Stream ciphers process messages one bit or byte at a time when en/decrypting – need not wait the entire block

• DES (Data Encryption Standard) cryptosystem

• Controversy over design – in choice of 56-bit key (vs Lucifer 128-bit) – design criteria (of the S-boxes) were classified

• DES is a single combination of these techniques (a substitution followed by a permutation) on the plaintext

Round-key is 48bits

• DES shows strong avalanche effect– to make attacks based on guessing difficult • S-boxes are non-linear (substitution) – provides confusion

3DES (Triple DES)– to preserve the investment in DES – for quicker deployment

• Three stages of DES – with two different keys • some attacks are possible but impractical

• E-D-E sequence C = EK3[DK2[EK1[P]]] PGP, S/MIME

• Double DES –meet-in-the-middle attack (which is a known-plaintext attack) • complexity of this attack is close to the complexity of the single DES brute-force attack, so double-DES is useless

• Blowfish • Developed by Bruce Schneier GnuPG, SSH

• RC family • Set of symmetric-key encryption algorithms invented by Ron Rivest

• Ron’s Code 5 • highly parametric • word oriented processing

• IDEA • International Data Encryption Algorithm

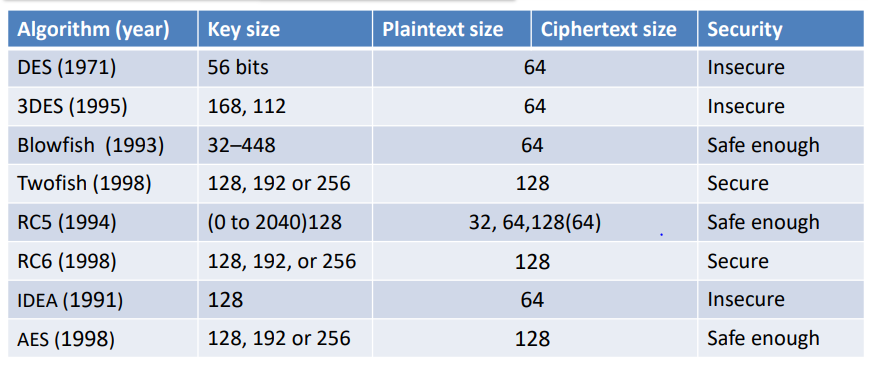
• AES Rijndael (from Belgium)

AES Requirements • private key symmetric block cipher • 128-bit data (block size) • 128/192/256-bit keys • stronger & faster than Triple-DES • active life of 20-30 years • provide full specification and design details

5 AES candidates • MARS (IBM) • RC6 (USA) • Rijndael (Belgium) • Serpent (Europe) • Twofish (USA)

– general security– software implementation performance– hardware implementation–attacks

Characteristics – resistant against known attacks – speed and code compactness on many platforms – design simplicity



Modes of Operations

1) Electronic Codebook (ECB) Mode

• each block is encrypted independent of the other blocks – using the same key • not so secure for long messages due to repetitions in code • Notation: C = E(P,K) P = D(C , K) • Trudy cuts and pastes, Same plaintext yields same ciphertext!

2) Cipher Block Chaining (CBC)

P xor IV/C -> E[K] = C, D[C, K] xor IV/C -> P

• each previous cipher blocks is XORed with current plaintext • each ciphertext block depends on all previous blocks • need Initialization Vector (IV) known to sender & receiver, Same plaintext yields different ciphertext!

3) Cipher FeedBack (CFB)

E[IV, K] of s bits xor P = C, E[IV, K] of s bits xor C = P

• **damaged packet affect only next block**

4) Output FeedBack (OFB)

nonce (means "used only once") • **damaged packet affect only next block**

**E[None+K] xor P = C, E[Nonce+K] xor C = P**

5) Counter (CTR)

• For the same key, the counter value should not repeat – same problem as in OFB • efficient – can do parallel encryptions

E[Counter , K] xor P = C, E[Counter, K] xor C = P

Random Numbers

– nonces in authentication protocols to prevent replay – session keys – public key generation – keystream for stream ciphers • Characteristics of random numbers – Statistical randomness

Pseudorandom Number Generators & Psuedorandom Functions

– PRNG is mostly context independent while PRF is context dependent

• When used in a cryptographic operation, seed must be kept secret

PRNG/PRF Requirements

– Uniformity: the occurrence of zeros and ones must be equally likely

– Scalability: any subseqeunce must pass randomness tests as well

– Consistency: must not dependent on a particular seed value

- Unpredictability

• Characteristics of the seed– so must be random or pseudorandom number

Linear Congruential Generator

• Common iterative technique using: Xn+1 = (aXn + c) mod m X0 is the seed

– function generates a full-period– generated sequence should appear random

So, not a secure mechanism

Using Block Ciphers as PRNGs

• for cryptographic applications, can use a block cipher to generate random numbers • often for creating session keys from master key • Standard methods • CTR Xi = EK[V+i] • OFB Xi = EK[Xi-1] X0 = EK[V]

Stream Ciphers

• process the message bit by bit

– A PRNG should eventually repeat • long period makes cryptanalysis difficult – statistically randomness– large enough key (128-bit would be good to guard against brute-force attacks)

• randomness of keystream destroys any statistical properties in the message – as in Vernam cipher and one-time pads

• Better than block ciphers in terms of – code space– throughput • but must never use the same keystream more than once

• are useful if data are transferred as a stream – web browser – voice – video • actually any block cipher can be used as a stream cipher – CFB mode of operation (and OFB and CTR )

RC4 • Ron’s Code 4

• variable key size, byte-oriented stream cipher • simple but effective • widely used (SSL/TLS, WEP/WPA) • Some attacks reported, but not practical for key size greater than 128-bit

**4. Public-Key Cryptography**

– ((a mod n) + (b mod n)) mod n = (a + b) mod n

– ((a mod n)(b mod n)) mod n = ab mod n

• Additive inverse of x mod n, denoted –x mod n, is the number that must be added to x to get 0 mod n

• Multiplicative inverse of x mod n, denoted x -1 mod n, is the number that must be multiplied by x to get 1 mod n

• x and y are relatively prime if they have no common factor other than 1 • x -1 mod y exists only when x and y are relatively prime

• ϕ(n) is “the number of numbers less than n that are relatively prime to n” – Here, “numbers” are positive integers– ϕ(p) = p-1 if p is prime – ϕ(pq) = (p-1)(q-1) if p and q prime

• Public-Key Cryptography – General Characteristics

• uses 2 Keys are related to each other but it is not feasible to find out private key from the public one

• based on number theoretic hard problems

• 3 misconceptions about PKC – it replaces symmetric crypto– PKC is more secure– key distribution is trivial in PKC since public keys are public

Why Public-Key Cryptography?

key distribution

you still need trusted third parties – digital signatures (non-repudiation)

Applications of Public- Key Cryptosystems– encryption/decryption – digital signatures – key exchange

RSA all/DH only key exchange

• Some Issues of Public Key Schemes

• like private key schemes brute force attack is always theoretically possible – use large keys – consider the security vs. performance tradeoff

• due to public key / private key relationships, number of bits in the key should be much larger than symmetric crypto keys – to make the hard problem really hard – 80-bit symmetric key and 1024-bit RSA key has comparable resistance to cryptanalysis

• a consequence of use of large keys is having slower encryption and decryption as compared to private key schemes – thus, PKC is not a proper method for bulk encryption

• Public-Key cryptosystem : RSA

• by Rivest, Shamir & Adleman of MIT in 1977 – published in 1978 • uses large integers – 1024+ bits • security depends on the cost of factoring large numbers

C= Me mod n

P= Cd mod n

Why RSA Works

• Euler's Theorem: a ø(n)mod n = 1 where gcd(a,n)=1

e.d=1+k.ø(n) for some k

• C d = M e.d = M1+k.ø(n) = M1.(M ø(n)) k = M1.(1)k = M1 = M mod n

Computational Aspects

– modular exponentiation for encryption and encryption – primality tests – finding inverse of e mod Φ(n)

• RSA Security

• 4 approaches of attacking on RSA

– brute force key search • not feasible for large keys • actually nobody attacks on RSA in that way

– mathematical attacks

• 3 forms of mathematical attacks –find **p.q,** **ø(n)** • so RSA cryptanalysis is focused on factorization of large n

• increase in computational power • biggest improvement comes from improved algorithm

– side-channel attacks

– based on timing variations in operations

– some operations are slow, some faster depending on the key

– use constant exponentiation time – add random delays – blinding (offered by RSA Inc.)

– chosen-ciphertext attack • Some algorithmic characteristics of RSA can be exploited to get information for cryptanalysis

**5. Public-Key Cryptography II**

• Diffie-Hellman Key Exchange

• First PKC offered by Diffie and Hellman in 1976 • purpose is secure key-exchange – actually key “agreement” – both parties agree on a session key without releasing this key to a third party • Security is in the hardness of the discrete logarithm problem – given ab mod n, a and n, it is computationally infeasible to find out b if n is large enough prime number

D-H Key Exchange – PK Management

• Two issues

– should we use global parameters (α and q) fixed for all public keys or unique?

– do we need to make sure that a particular public key Yi produced by i?

• Anonymous public values are problematic – causes man-in-the-middle attacks

– Solution: public values and parameters should be either known or should be endorsed by a trusted entity

• previous example of trusted database is one solution • public key certificates are the most common solution

• Certification Authority (CA): binds public key to particular entity, E. • E registers its public key with CA. – E provides “proof of identity” to CA. – CA creates certificate binding E to its public key. – certificate containing E’s public key digitally signed by CA: CA says “This is E’s public key.”

• when Alice wants Bob’s public key: – gets Bob’s certificate (Bob or elsewhere). – apply CA’s public key to Bob’s certificate, get Bob’s public key

• Hash Functions – General idea

• are used to generate fixedlength fingerprints of arbitrarily large messages– H(M) calculations should be easy and fast • indeed they are even faster than symmetric ciphers

Requirements and Security

• Hash function should be a one-way function – given h, it is computationally infeasible to find x such that h = H(x) – complexity of finding x out of h is 2n , where n is the number of bits in the hash output – Called one-way property (a.k.a. preimage resistance)

• Weak collision resistance (a.k.a. second preimage resistance) – given x, it is computationally infeasible to find y with H(x) = H(y) – complexity of attack is 2n

• (Strong) collision resistance – It is computationally infeasible to find any pair x, y such that H(x) = H(y) – complexity is 2n/2

• Ralph Merkle – a sequence of compressions– MD5, SHA-1, SHA-2 and some others are based on that idea

• MD5 – Message Digest 5 – another Ron Rivest contribution – arbitrarily long input message • block size is 512 bits – 128-bit hash value– brute force attacks • 264

• SHA-1– input size < 264 bits – hash value size 160 bits, 280 is not-a-bad complexity, several other attacks are reported

• SHA-256, SHA-384 and SHA-512 – for compatible security with AES (SHA-224) is later added

• SHA-3 Sponge construction

• Digital Signatures • Mechanism for non-repudiation

**Generally signatures are created and verified over the hash of the message – Why?**

Digital Signature – RSA approach

M+E[H(M), PRA], H(M)==D[Sig, PUA]

Digital Signature – DSA approach

M+E[H(M)+PUG+PRA+K], D[H(M)+PUG+PUA+S+R]==R

Collision resistant hash functions and digital signatures

• Have you seen the reason why hash functions should be collision resistant? – because otherwise messages would be changed without changing the hash value used in signature and verification

• Birthday attack 2n/2 where n is the hash size, hash could be replaced with the legitimate one without affecting the signature

• Message Authentication Code

Message Authentication also covers integrity

Digital Signatures – provides integrity + authentication + nonrepudiation

Mechanisms for Message Authentication

1) Message Encryption – provides message authentication, but … • Provides encryption. What about authentication, optional • What about public-key encryption? • Provides confidentiality, but not authentication

2) Message Authentication Code Functions – similar to encryption functions, but not necessarily reversible

• Is MAC a signature? – No, because the receiver can also generate it

Hash based Message Authentication

-with confidentiality uses two key, one for MAC, one for encrypt/decrypt

M+E[H(M), K], H(M)==D[MAC, K]

E[M+H(M), K], D[(M+H(M)), K], H(M) == H(M)

-without confidentiality

M+C(M, K), C(M, K) == C(M, K)

HMAC: IPSec and SSL

• uses hash function on the message: HMAC = Hash[(K+ XOR opad) || Hash[(K+ XOR ipad)|| M)]]

• overhead is just 3 more blocks of hash calculations than the message needs alone • any hash function (MD5, SHA-1, …) can be used

HMAC Security

• HMAC assumes a secure hash function, it has been proved that attacking HMAC is equivalent the following attacks on the underlying hash function – brute force attack on key used – birthday attack • (2n/2 messages)

**6. User Authentication**

• Passwords Usability problems

• Forgotten passwords might not be recoverable

• Entering passwords is inconvenient

• If password is disclosed to unauthorized individual, the individual can immediately access protected resource • Unless we use multi-factor authentication

• If password is shared among many people, password updates become difficult

• Attacks on Passwords Keystroke logging, Shoulder surfing, Interface illusions / Phishing

• Password related threats

1. Password Guessing • Exhaustive Search (Brute Force) • Intelligent Search

How the system helps?

• Password ageing • Limit login attempts • Use of CAPTCHA • Inform user

2. Password Spoofing

– fake login prompts

• remote login is even worse, – telnet sends passwords in clear – use SSH (Secure Shell)

• Shoulder surfing – Check surroundings in public spaces

3. Cracking the password file

Password Storage

• Passwords are generally stored in encrypted form – using symmetric encryption or one-way hash functions

• Possible off-line attack dictionary attacks passive off-line attack • unsuccessful attempt limits do not help

• Hash password with salt

How to prevent dictionary attacks on password files

• Slow down password encryption

• Do not make the password file publicly readable

• Password Salting slows down dictionary attack • if two users have the same password, their encrypted passwords will not be same

• Biometric Authentication Approaches Fingerprint, vein geometry, retina pattern, palm print, face

– does not have 100% accuracy • false accept • false reject

• if copied or broken, you cannot change it • people may not like their fingerprints are taken as criminals or beams in their eyes

• What you have – smartcards and smart tokens are the best examples

– can be stolen or lost • should be used together with a PIN or password

• What you do– dynamic signatures • pressure, speed, orientation are properties as well as the shape – Keyboard typing • speed, intervals between keystrokes – false accept, false reject problems exist here too

• Two Factor Authentication

**7. Network Security (IPSec protocol)**

• Internetwork Protocol (IP) – provide interconnection across different network

• IPv4 32 , IPv6 128

App – tcp – ip – ethernet

App-transport-network-network interface

App-tcp-ip-llc-mac-physical

Is IP Secure?

• Content (Payload) is not encrypted – confidentiality is not provided

– IP sniffers are available on the net • IP addresses may be spoofed

– authentication based on IP addresses can be broken

Where to provide security?

• Application-layer? – S/MIME, PGP – email security – Kerberos – client / server – SSH – secure telnet

• Transport level? – SSL / TLS – between TCP and Application

• IP level – IPSec

• IPSec overview

• provides authentication and confidentiality at IP level – also has key management features

• Applications – VPNs (Virtual Private Networks)

• Interconnected LANs over the insecure Internet

• IPSec support is mandatory for IPv6 products, optional for v4

Benefits of IPSec

• in a firewall/router, IPSec provides strong security to all traffic entering the network – without passing the security overhead to the internal network and workstations

– user transparent: no need to assume security-aware users, no per-user keys

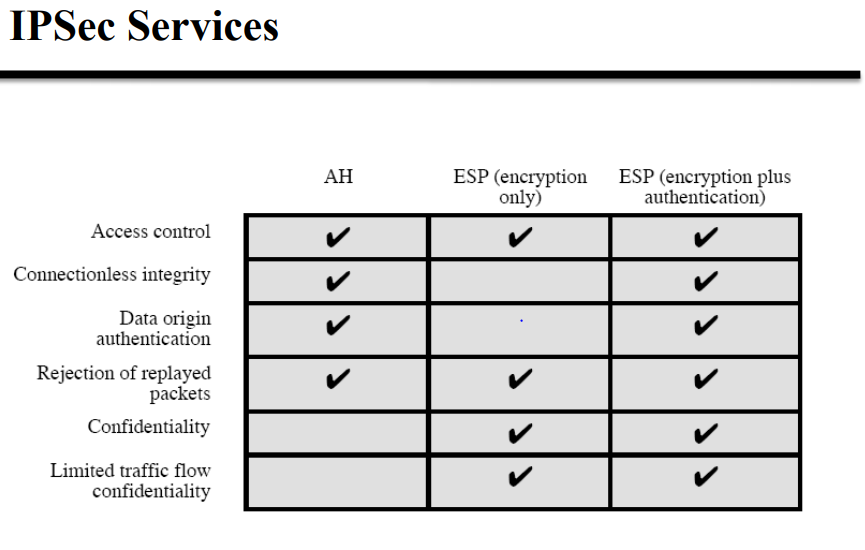
• IPSec is below transport layer – No need to upgrade applications when IPSec is used, if IPSec is implemented and configured in user machines

• IPSec Protocols

• Authentication Header (AH) – defines the authentication protocol – no encryption

• Encapsulating Security Payload (ESP) – provides encryption – optionally authentication

• Key distribution and management are also in different RFCs



Security Associations (SA)

• a one-way relationship between sender & receiver – specifies IPSec related parameters

• Identified by 3 parameters: – Destination IP Address – Security Protocol: AH or ESP – Security Parameters Index (SPI) • There are several other parameters associated with an SA – stored locally in Security Association Databases (SAD)

SA Parameters (some of them)

– Sequence Number Counter – Anti-replay window • AH info – authentication algorithms, keys, key lifetimes, etc. • ESP info – encryption (and authentication) algorithms, keys, key lifetimes, etc. • Lifetime of SA • IPSec Mode: Transport or Tunnel

• Security Policy Database (SPD) is used to assign a particular IP traffic to an SA – fields of an SPD entry are called selectors

• Outbound processing – send

• Inbound processing – recieve

• IPSec Modes

• Transport Mode – security is basically for the IP payload (upper-level protocol data) – IP header is not protected (except some fields in AH) – Typically for end-to-end communication

• Tunnel Mode – secures the IP packet as a whole incl. header(s) – actually puts all IP packet within another (outer)– Typically for router-to-router, or firewall-to-firewall communication

Authentication Header (AH)

• Provides support for data integrity and authentication of IP packets • Authentication is based on use of a MAC – parties must share a secret key • in SA

SPI+Sequence number+Authenication data:ICV

AH – Anti-replay Service • Sequence numbers, receiver side window

AH – Transport Mode == AH – Tunnel Mode

Encapsulating Security Payload (ESP)

• provides – message content confidentiality • via encryption – limited traffic flow confidentiality and measures for traffic analysis • by padding • by encrypting the source and destination addresses in tunnel mode – optionally authentication services as in AH • via MAC (HMAC), sequence numbers

SPI+Sequence numver+encrypted data+ICV

Padding in ESP

• several purposes and reasons – encryption algorithm may require the plaintext to be multiple of some integer n , to provide partial traffic flow confidentiality by concealing the actual length of data

Transport Mode ESP

• transport mode is used to encrypt & optionally authenticate IP payload– data protected but IP header left in clear – so source and destination addresses are not encrypted – Mostly for host to host (end-to-end) traffic

Tunnel Mode ESP

• Encrypts and optionally authenticates the entire IP packet – add new (outer) IP header for processing at intermediate routers, traffic analysis can somehow be prevented – good for VPNs, gateway to gateway (router to router) security

Combining Security Associations

• SAs can implement either AH or ESP – Solution1: use ESP with authentication option on – Solution2: apply ESP SA (no auth.) first, then apply AH SA – Solution3: Apply AH SA first, then ESP SA • encryption is after the authentication

Key Management in IPSec – generate and manage SAs for AH and ESP – asymmetric

IKE Key Determination• Key exchange protocol based on Diffie-Hellman

ISAKMP • Internet Security Association and Key Management Protocol SA management protocol SA establishment protocol run in ISAKMP

**8. Software Security**

• Threats people who are able to take advantage of security vulnerabilities to attack systems. Also known as adversaries.

Vandals

Hacktivists attack systems for political goals. to spread their message

cyber, criminals Focus on monetizing information Specialists who sell services to other criminals

cyberspies Threats that work for a nation state Cyberespionage and cyberwarfare

Insiders are threats who are members of the organization that they are attacking. Insiders are dangerous because

Vulnerabilities are weaknesses in a system that allow a threat to obtain access to information assets in violation of a system’s security policy.

Attacks are actions taken by threats to obtain assets from systems in violation of the security policy

• Threat Models describes which threats exist to a system, their capabilities, resources, motivations, and risk tolerance. Also known as an adversary model.

• Motivations • Money • Espionage • Fame/status • Learning • Entertainment • Hacktivism • Sabotage • Terrorism

• Intent the goal of the attacker

• Resources • Skilled personnel • Money • Computational power • Technology • Infrastructure

• Capabilities Computational Informational Access

• Risk Aversion is a tendency to avoid taking actions with negative consequences. arrested, imprisoned, fined. – Physical attacks Attacks from within the country of target are riskier, Attacks from a country with an extradition agreement with the country of the target are riskier

• Access

– Insider with administrative privilege. – Insider with privilege to access the desired target. – Insider with ordinary user level access. – Backdoors from previous attacks on same target. – No access

Advanced Persistent Threat (APT) – Sophisticated – Targeted. – Skilled personnel. – May be backed with considerable budget.

• Attacks is an action taken by a threat to gain unauthorized access to information or resources or to make unauthorized modifications to information or computing systems.

How are Digital Attacks Different? Automation Action at a Distance Technique Propagation

Spoofing is when a threat masquerades as another entity on a telecommunications network.

Sniffing Packet is when a program records wired or wireless network packets destined for other hosts.

– Wireless traffic– Antennas Wired traffic Sniffing used to – Obtain passwords. – Obtain other confidential information

Man in the Middle an active eavesdropping attack, in which the attacker connects to both parties and relays messages between them.

Injection Attacks send code to a program instead of the data it was expected, then exploit a vulnerability in the software to execute the code. – Buffer overflows

Denial of Service A denial of service (DoS) attack attempts to make computer or network resources unavailable to its intended users. A distributed DoS (DDoS) attack

Account Compromise Attackers can take over a user’s account and use that account’s permissions to obtain or modify data. Guessing attacks with automated software. – Reuse of passwords exposed in a data breach. – Phishing. – Keylogging. – Password resets. Attackers can temporarily compromise an attack by hijacking a user session via a MITM attack.

Social engineering is the psychological manipulation of people to reveal confidential information or perform actions to violate security policy.

Web applications are subject to a variety of attacks.

Attack Vector means of delivering an attack.

E-mail(attachments) Malvertising Network access Remote access systems like VPNs Social engineering Supply chains

Attack surface: the set of ways an application can be attacked. Used to measure attackability of app.

Why Attack Surface Reduction? If your code is perfect, why worry? – All code has a nonzero probability of containing vulnerabilities. – Even if code is perfect now, new vulns arise.

Attack Trees are a way to model possible attacks against a specific target or asset.

• Exploits is a technique or tool that takes advantage of a vulnerability to violate an implicit or explicit security policy.

1. The type of vulnerability they exploit. 2. Local or remote. 3. Result of exploit

Indicator of Compromise

• Malware software designed to gain access to confidential information, disrupt computer operations, and/or gain access to private computer systems.

– Trojan Horses 트로이 목마는 정상 프로그램으로 가장하지만 악성 지침에 포함되어 있다. 트로이 목마가 작동하려면 반드시 피해자가 실행시켜야 한다.

– Viruses when executed, replicates by inserting copies of itself into other files. This process is called infecting.

– Worms spreads itself to other computers. No execution

– Ransomware 데이터를 암호화하여 볼모로 잡고 돈 낼 때까지 기다리는 악성코드 프로그램

- Information Stealers target specific types of information

– Spyware and adware keyloggers software, mobile spyware, computer spyware

– Backdoors - 정상적인 인증 없이 시스템 접근을 하는 방법

– Rootkits • Execution Redirection • File Hiding • Process Hiding • Network Hiding • Backdoor

– Botnets 다른 컴퓨터 감염시켜서 내가 원할 때 원하는 대로 행동하게 하기

Covert channels enable communication using techniques not meant for information exchange.

Vulnerabilities in PC, Server, Mobile, Embedded, Third party software

Mitigation is a process, technique, tool, or software modification that can prevent or limit exploits against vulnerabilities. – A password length policy A firewall Checking for the lock icon in the location bar of your browser

Security Patches is a software modification designed to prevent or limit a vulnerability.