

### Udemy Nathan cyber security basic

<https://www.stationx.net/canarytokens/>

### Basic theory - asset and vulnerability

Protect confidential, not afford to lose, irreplaceable, cost most valued damages, impact reputation

privacy(confidential), anonymity (identity hiding), Pseudonymity (false identity, such as bitcoin)

security and vulnerabilities (ssh, https, 2FA, vpn etc)

threats (virus malware, hacking, spyware, rootkits, adware, phishing, vishing, exploit kits)

adversaries ( hacker, cyber criminals, spies, crackers, law enforcement governments)

assets and protection, granularity, risk assessments

risk = vulnerability X threats X consequences, trade off risk and beneficial

select - implement - assess - monitor

security vs privacy vs anonymity: conflict

confidentiality(keep your self), integrity ( accuracy unmodified) availability ( functional): CIA Triad

Defense in depth: prevention- detection - recovery

Zero trust model, the less trust, including yourself, the safer. trust nothing trust nobody. never put online. nothing is safe online

### basic theory - current threat and vulnerability

Value of hack: not a person, but a Bot -automate AI software to continuous attacking you.hijacking

Top 3 things need to stay safe online?

security bugs:always exist, as human writing code.OS, firmware, app, web browser ( js, java), known bug + patches, unknown bugs + zero days , no patch

<https://www.cvedetails.com/>

<https://exploit.db.com/>: public available, patch available, can be used to exploit unpatched system

### basic theory - current threat and vulnerability (cont)

hacker-white (ethnic legal hacking) and black hacker (cyber criminals)

cracker- crack the key of a software

cyber criminal (black hacker,

malware:macro virus,stealth virus, polymorphic virus,self-garbling, bots and zombies, worms OS rootkit(bed in kernel), firmware Rootkit,key logger, trojan, Remote access tool (RAT)

Ransomware: designed to deny access to a computer system or data until a ransom is paid, usu by phishing.

spyware(spy), adware(formal spyware, highjack web searching), browser hijacking, scareware( fake info to scare you to pay), pup ( potentially unwanted programs)

phishing(trick you to click, easy and high successful rate, 30% people still be fooled, email is common way to phishing): google.xx-xxx.com, check HLD high level domain, goog1e, g00gle, hidden URLs

`<a href = ....>fake link </a>`

vishing: phone/voice

smsmhing: sms

spamming: unsolicited message, email, message etc. minimal cost, high earning.

doxing:ren-rou, googling/anything to get info for some body social engineering: - scams, cons, tricks, fraud

cpu hijackers: crypto mining malware and cryptojackers

darknet (only accessible with special tools) vs clearnet(google, amazon)

dark market: access through darknet

exploit kits

government, spies, and secrete stuff: 5 eyes

regulating encryption, mandating insecurity and legalizing spying

trust and backdoors: formal methods, closed, open source, binaries, hash,digital signature

censorship



### encryption

plan text -> cipher text -> plan text: encryption (cipher) decryption (decipher)

algorithm: public/lock

key: secret/password

windows: encryption method 256, 128, legacy (zip 2.0) key length and key space

AES (Advanced Encryption Standard): symmetric algorithm (uses 1 Key private) password becomes the key

DES (Data Encryption Standard), 3 DES (triple-DES), Blowfish, RC4-6 brute force, dictionary force, hybrid the two

Asymmetric encryption: 2 keys (public and private),

RSA (Rivest Shamir Adleman), ECC (Elliptic Curve Cryptosystem): digital signature; D (Diffie Hellman), El Gamal

Key exchange and agreement: encrypt with one, decrypt with another.

Confidentiality (private key): decrypt with receiver's private key

authentication (private key): encrypt with sender's private key-signature

Non-repudiation

integrity:

Asymmetric better key distribution, scalability, authentication and non-repudiation (not deniable), slow, mathematically intensive

Symmetric: fast, strong

RSA: AES (1024-80, 2048-112, 3072-128, 15360-256)

hybrid cryptosystem: RSA (key distribution/encapsulation) and AES (data encryption encapsulation)

HTTPS: handshaking (client hello/server hello) --> exchange certificate (authentication, authorized server issues their private key encrypted certificate, name, domain, public key of server user, client is not required in general) --> exchange key (server public key, client symmetric key)

<https://robertheaton.com/2014/03/27/how-does-https-actually-work/>

<https://nickfishman.com/post/50557873036/reverse-engineering-native-apps-by-intercepting-network>

<https://www.hostinger.com/tutorials/ssh-tutorial-how-does-ssh-work>

### SSL TLS

Secure Sockets Layer (SSL) -> Transport Layer Security TLS (1.0-1.3)

Confidential (symmetric key AES), authenticated (public key, digital signature) and integrity (integrity check, hash value etc)

cipher suites setup on server

Session Encryption Negotiation

1 shared larger prime number

2 AES algorithm

3 generated primes as private key

4 private key + shared prime + AES > public key > distribute to another party

5 private key + shared prime + public key of the other > shared symmetric key, generated independently but the same

6. the shared key is used to encrypt the connection

Authentication:

client password and username or SSH Key Pairs.

<https://www.hostinger.com/tutorials/ssh-tutorial-how-does-ssh-work>

<https://www.hostinger.com/tutorials/ssh/how-to-set-up-ssh-keys>

### Https

HTTPS: handshaking (client hello/server hello)

--> exchange certificate (authentication, authorized server issues their private key encrypted certificate, name, domain, public key of server user, client is not required in general)

--> exchange key (client sends symmetric key)

SSH stripping: client > http > middle man > https > server using Kali, or hardware

avoid: https only, tunneling (VPN/SSH), only trusted website

sniffdet

arpwatch

VLAN: virtual LAN

ssllabs.com

SNI server name indication

<https://robertheaton.com/2014/03/27/how-does-https-actually-work/>

<https://nickfishman.com/post/50557873036/reverse-engineering-native-apps-by-intercepting-network>

VLAN: virtual LAN



### Hash

Integrity: hash function, checksum  
MD5, Sha-256  
powershell: get-filehash -Algorithm Sha512 c:\test.txt.  
download checksum comparison: verify download  
hash password to save d to use as verification, original password not saved.  
HMAC: haseh based msg authentication code

### digital signature

hash algorithm - hash value --> sender private key - signed msg  
authentication, nonrepudation, integrity  
signed msg --sender's public key -hash value  
windows device guide

### digital certfictes and https

digital signature from well know trusted company( third parties)  
local library of digital cert library auto loaded (roots)  
local digital cert manually loaded(self signed, trusted parties)  
compromised/fake digital cert: really risk  
CA Ecosystem  
CA example mistakes  
SSL sniff  
CA patrol  
cert fingerprints  
pinning

### E2EE, steganography

E2EE end to end encryption:PGP ZRTP OTR SSL/TLS  
use E2EE always possible  
steganography: the practice of concealing a file, message, image, or video within another file, message, image, or video.  
openpuff

### Setup testing environments

type2 Hosted: hard ware ->OS ->hypervisor->OS  
type 1 native: hardware >hypervisor->OS  
vmware or virtualbox  
testing environment |security options  
install virtual OS: physical DVD, virtual DVD (ISO), prebuilt virtual disk/image (.ova form virtual box)  
Kali: debian, 600 penetration testing tools.  
osboxes.org for prebuilt images

