Co-funded by the Erasmus+ Programme of the European Union

# InCyT træningsmodul

| | |
|---|---|
| **Informationssikkerhed for medarbejdere<br>Enhed 2 Uge 4<br>Enhedens navn: Malware og mest almindelige angreb** | |
| ***Læringsaktiviteter*** | ❑ **Webinar** ❑ **Øvelser** ❑ **Værksted** ❑ **Præsentation** ❑ **Andet** |
| ***Læringsansvarlig*** | Claudio Fornaro |
| ***Læringsaktivitetsmål*** | 1. Introduktion til de forskellige typer malware<br>2. Kendskab til de mest almindelige angrebstyper |
| ***Færdigheder, der skal opnås*** | • Grundlæggende viden om, hvad der er malware og de mest almindelige angreb på computersystemer<br>Soft Skill 2 - SS2 |
| ***Varighed af læringsaktivitet*** | En uge |
| ***Læring krav*** | ***Hvad skal der til?*** |
| | • Der kræves ingen forudgående specifik viden |

Co-funded by the
Erasmus+ Programme
of the European Union

## Kort information om modulet

Når det kommer til malware, antyder navnet allerede noget ondsindet. Dette udtryk refererer generelt til software, der er designet med vilje til at forårsage skade af forskellig art, især (men ikke begrænset til): blokering af tjenester og kommunikation, aflytning af transmissioner, tyveri af private oplysninger (personlige, virksomhedsmæssige eller offentlige) og personlige identiteter, uautoriseret adgang til information eller systemer, fratagelse af brugere af adgang til information.
Efter en indledende del skitseres de grundlæggende begreber for malware, sammen med beskrivelsen af hovedtyperne, de risici, de medfører, og hvordan disse kan afbødes.
Anden del af modulet har til hensigt at give et overblik over de mest almindelige angreb, som et system kan lide, denne liste er på ingen måde komplet, da nye typer angreb opdages hver dag. Disse angreb er forsøg på at få adgang til et computersystem eller forstyrre dets tjenester. Beskrivelsen starter fra de mest almindelige og nemmeste angreb (Phishing) til mere komplekse: Password Attacks, Buffer Overflow, SQL Injection, Cross-site Scripting (XSS), Cross-Site Request Forgery (CSRF), Fil-relaterede angreb, Denial of Service (DoS) og Distributed Denial of Service (DDoS), Man in The Middle (MITM), ARP-spoofing eller ARP-forgiftning, DNS-spoofing, Drive-by-angreb,
Genbrug af legitimationsoplysninger, kapring af websession, kapring af TCP-sessioner, IP-spoofing, Zero-day-udnyttelse og konsekvenserne af kodeudslip.

## Lærematerialets indhold

Inden for datalogi er "hacker" og "cracker" to udtryk, der bruges til at indikere folk med stor viden om computersystemer. Den største forskel mellem de to ligger i den holdning, de handler med. En **hacker** (udtrykket blev født i anden halvdel af 1900-tallet på MIT) handler ved at udforske systemer for bedre at forstå, hvordan de fungerer, og forsøge at opdage nye fejl for at løse dem og dermed øge sikkerheden i et system. Nogle handlinger kan kun være demonstrative eller af etiske årsager, men aldrig for profit. En **cracker** , på den anden side, bruger sine evner ulovligt ved at forsøge at bryde ind i systemer for at tjene penge ved at stjæle data eller på en destruktiv måde.

Hackere forsøger konstant at tilegne sig ny viden om computersystemer for bedre at forstå dem eller forbedre deres funktioner, ydeevne og sikkerhed. Nogle af dem engagerer sig også i demonstrative og etiske handlinger, men aldrig for profit. Hackerhandlinger udføres næsten altid uden nogen tilladelse, så loven straffer dem (med store forskelle mellem landene), af denne grund gemmer hackere sig normalt bag kaldenavne. Nogle gange ansætter kloge sikkerheds- eller IT-virksomheder i stedet for at retsforfølge hackere dem ofte til at drage fordel af deres høje kompetencer og kreativitet (muligvis efter at have betalt en del af deres gæld til Justice). De bliver derefter ansat til at teste applikationer, sikkerhedssystemer og programmer både i produktion og

allerede på markedet for at blokere angreb og for at forhindre problemer, der kan opstå i fremtiden fra systemfejl.

*Rigtige* hackere har et ry som eksperter og er højt respekteret i deres samfund. Konkurrencer tilrettelægges med det formål at "pierce" et system eller program på kortest mulig tid. Disse konkurrencer kaldes *hackathons* og er ofte sponsoreret af computerindustriens giganter.

Udtrykket "cracker" tilskrives Richard Stallman (en velkendt figur fra fri softwarebevægelsen, skaberen af GNU-projektet og medlem af hackersamfundet siden 1970'erne). Han ønskede at undgå misbrug af begrebet "hacker" ved at skelne mellem, hvem der handlede uden at ville skade, og hvem der tværtimod handler for at have fortjeneste eller blot give skade. En cracker er en cyberkriminel, der forsøger at angribe computersystemer på en uautoriseret og ulovlig måde, med ond vilje, og forsøger at drage svigagtig fordel af dem ved at udnytte sårbarheder, der endnu ikke er kendt og løst. Crackere er ofte involveret i industrispionage, datatyveri og blokering af netværk og websteder, hærværk mv.

Begreberne hacker og cracker har subtile grænser, der adskiller dem. For eksempel kan en indtrængen betragtes som en ulovlig handling for nogle mennesker og ikke for andre, og Loven afspejler disse holdninger; men dette dokument handler ikke om etiske eller lovgivningsmæssige aspekter. I stedet må vi overveje, at de to begreber i dag desværre betragtes som synonyme.
**I dette dokument vil vi *desværre* bruge det mere almindelige og misbrugte begreb hacker for begge kategorier, idet vi er sikre på, at den kloge læser helt vil være i stand til at skelne mellem en *hacker* og en *cracker* .**

"Hackere" kan groft klassificeres i flere kategorier, de mest udbredte er følgende.

**Black hat hackere** bruger deres viden til ondsindede eller kriminelle formål, idet de passer på ikke at blive sporet.

**White hat hackere** tester systemer for at identificere mulige sårbarheder og eliminere dem for at forhindre uønsket adgang. De er ofte sikkerhedseksperter, der betales til dette formål.

**Gray hat hackere** ligner hvide hatte, men de udfører ofte forskning i systemsårbarheder uden autorisation, og når først disse sårbarheder er opdaget, bruger de dem til at tvinge leverandører til at løse problemet, normalt gratis. Nogle sælgere belønner dem med penge, andre retsforfølger dem.

**Selvmordshackere** er normalt mindre kompetente end sort/hvide hatte, men de handler af ideologiske årsager og har ingen interesse i at skjule deres identitet.

**Statssponserede hackere** betales af en regering for at iværksætte cyberangreb mod et andet land; de har adgang til store ressourcer som stillet til rådighed af staten samt beskyttelse mod

retsforfølgelse for de begåede lovovertrædelser. De handler for at trække fortrolige oplysninger, planer eller projekter fra eller for at skabe generaliserede lidelser og skade befolkningen, hæren, regeringsledelsen, hospitaler, anlæg til produktion af energi, vand og strømforsyning osv. Disse handlinger betragtes som ægte krig. handlinger.

**Scriptkiddies** har ikke eller har endnu ikke tilegnet sig store færdigheder, de gør normalt brug af scripts og værktøjer skabt af andre hackere. Alle hackere, før de tilegner sig store færdigheder, er script-kiddies.

**Cyberterrorister** er individer eller grupper med fjendtlige hensigter, der generelt har til formål at skabe kaos i et land; de ligner statssponsorerede hackere, men har (endnu) ikke en regering til at støtte dem, eller støtten er ikke tydelig.

## Malware

Malware er forkortelsen for " **mal** icious **software** ", et generisk udtryk, der identificerer ethvert ondsindet program, der er skadeligt for et system. Malware er den software hackere skaber for at forfølge deres mål: tjene penge, stjæle data, forstyrre aktiviteter osv.

En af de mest almindelige malware er den såkaldte en **trojansk hest** [1](eller blot **trojansk hest** ), et program, der virker nyttig software og ofte fungerer som sådan, men som har en skjult funktion, der udføres for at gøre sammen med den normale aktiveret, når applikationen er startede. Disse er ofte shareware-programmer eller crackede programmer, der distribueres modificeret for at bære den trojanske hest. I andre tilfælde er det bare malware med et misvisende navn, der får det til at virke som en nyttig software. Trojanske heste spredes ofte via e-mails eller ved at downloade programmer via internettet og spredes normalt ikke af sig selv til andre computere. Eksempler på ondsindet kode er *keyloggere* (programmer, der stjæler adgangskoder og sender dem til hackeren), *bagdøre* (programmer, der fungerer som en terminalserver og gør det muligt for fjernhackeren at få adgang til computeren på nogle måder uden om de normale godkendelsesprocedurer), *kryptominere* (programmer, der miner cryptocurrency), *bots* (forkortelse for robotter, bruges for eksempel til et distribueret lammelsesangreb).

**Rootkits** er software installeret af malware, der forsøger at skjule tilstedeværelsen af en malware i sig selv fra opdagelse; de opnår dette ved at ændre operativsystemerne.

Den malware, der inficerer anden software, kan være en virus eller en orm. Disse får deres navn fra den måde, de spredes på, snarere end fra nogen specifik type adfærd. En computervirus er **software** , der ligner trojansk hest, men kan producere kopier af sig selv ved at inficere (introducere sig selv

[1]Udtrykket er afledt af den antikke græske historie om den trojanske hest, der blev brugt til at invadere byen Troja ved stealth

i) anden software, inklusive operativsystemfilerne. Virussen spredes til andre systemer, når inficeret software kopieres og køres på disse systemer.

Svarende til en virus er en **orm** , dette er en selvstændig software, der fungerer på en computer og inficerer andre computere via netværket, men uden at inficere filer.

En **Ransomware** er en type malware, der begrænser adgangen til den enhed, den inficerer, hvilket kræver, at der betales løsesum for at fjerne begrænsningen. De vigtigste typer af ransomware er 1) *Skærmlåsende ransomware* , der blokerer skærme med en skræmmende anklage om ulovlig adfærd og beder ofrene om at betale et gebyr; 2) *Krypteringsbaseret ransomware* , hvor den ondsindede software en eller anden type filer (billeder, dokumenter) eller alle filer på en inficeret maskine. Når den fulde kryptering er færdig, informerer en pop-up normalt brugeren om, at (alle) filer er blevet krypteret, og at der skal betales et gebyr for at gendanne dem, normalt i kryptovaluta. Eksempler på krypteringsbaseret ransomware er CryptoLocker og WannaCry. En variant kendt som en **Wiper** ligner en ransomware, men ødelægger bare dataene, selvom en betaling ofte er påkrævet.

**Greyware** er en anden kategori af malware: Disse programmer kan forværre computernes ydeevne og kan forårsage sikkerhedsrisici, opføre sig på en irriterende eller uønsket måde, men er normalt mindre farlige (hvilket er en relativ opfattelse) end de tidligere kategorier af malware. Spyware (f.eks. keyloggers, programmer, der registrerer alle de taster, der trykkes på tastaturet), adware (reklamevinduer på skærmen eller i browseren, svigagtige opkaldere osv.), Potentielt uønskede programmer (PUP'er, applikationer, som brugeren installerede blot pga. en fejl i et aftalevindue under en anden programinstallation) er typiske former for gråvare. De kan være selvstændige programmer eller plugins til andre programmer, for eksempel browsere. Nogle gange begrænser de sig selv til at ændre indstillingerne for andre programmer, for eksempel for at bringe brugeren til et bestemt annoncewebsted eller vise en annonceside, hver gang browseren startes eller lejlighedsvis.

## Antivirus

Da antivirus-motorer forbedrer deres evne til at opdage malware, bruges en kombination af forskellige teknikker til at forsøge at undgå opdagelse og analyse. Nyere malware er i stand til at vedtage sofistikerede modforanstaltninger såsom:

- analysere og detektere ved fingeraftryk af miljøet for at vedtage specifikke skjulemetoder;
- forsøger at tilsløre sig selv for at forvirre antivirusdetektion baseret på malware-signatur;
- ved at anvende en timing-baseret unddragelse for at narre opdagelse af unormal adfærd, er malwaren normalt i ro og aktiverer sig selv i meget specifikke perioder (f.eks. under opstartsprocessen) eller efter bestemte handlinger foretaget af brugeren.

En meget svær at opdage teknik til unddragelse er *Fileless malware* eller *Advanced Volatile Threats* . Denne form for malware kører i hukommelsen og kræver ikke en fil for at fungere. Det bruger eksisterende systemværktøjer til at udføre sin aktivitet. Fraværet af nogen fil på filsystemet gør dem ret vanskelige at opdage og analysere, bortset fra, delvist, for de overvågningsværktøjer, der kontrollerer ressourceforbruget og opførselen af systemet. Stigningen af denne type angreb er meget høj.

**Risici**

Sårbar software kan udnyttes til at opnå privilegier eller omgå forsvar, indtil problemet er blevet rettet; dette er hovedårsagen til, at software skal være opdateret, især operativsystemet. Firewalls og Intrusion Detection Systems overvåger trafikmønstre på LAN.

Reglen om det mindste privilegium siger, at brugere og programmer ikke bør have flere privilegier, end de kræver, ellers kan malware drage fordel af dette.

**Afbødning**

Antivirussoftware blokerer og fjerner muligvis nogle typer malware, de kan give realtidsbeskyttelse mod installation af malware-software på en computer og en fuld scanning af filer og konfiguration (f.eks. Windows-registrering) på udkig efter potentielle trusler.

Personlige firewalls analyserer enhver netværksforbindelse og kan opdage usædvanlig aktivitet.

Sandboxing kører en applikation isoleret fra resten af systemet, nogle applikationer kan køre en specifik applikation, mens en virtuel maskine i mere komplekse systemer kunne være mere passende.

Webstedssårbarhedsscanninger udføres af nogle antivirussoftware mod en liste over velkendte farlige websteder og forhindrer forbindelsen til dem.

Netværksadskillelse, der opdeler et netværk i dele og kun muliggør den trafik mellem dem, der vides at være legitim, forhindrer skadelig software i at sprede sig over netværket. Moderne softwaredefinerede netværksteknikker er i stand til at implementere denne adskillelse.

"Air gap" isolation eller "parallel netværk" bringer netværkssegregation til grænsen ved fuldstændig at isolere en del af et netværk og give forbedret kontrol over ind- og udgang af software og data fra omverdenen. Selv denne løsning er ikke den endegyldige, da software og data skal ind i det isolerede system, for eksempel til patching. Et velkendt eksempel på brud på et isoleret netværk er den berømte **Stuxnet -** malware, der blev introduceret til målmiljøet via USB-drev "tilfældigt" glemt i nærheden, samlet op af fabrikkens personale og sat ind i fabrikkens computere for at se indholdet. Stuxnet har tre moduler: en orm, der styrer angrebet; en linkfil, der automatisk udfører ormen, når

den indsættes i en Windows-computer; og en rootkit- komponent, der er ansvarlig for at skjule alle ondsindede filer og processer for at forhindre detektion. Andre måder at krydse luftspalter på analyserer elektromagnetiske, termiske og akustiske emissioner.

## Mest almindelige typer angreb

Denne del beskriver nogle af de mest almindelige angreb på et computersystem. Det er ikke og kan ikke være komplet, da mangfoldigheden af angreb er ekstremt stor, hver enkelt har mange varianter og mange angreb og varianter bliver skabt og opdaget (og løst) hver dag. Det betyder, at formålet med denne del ikke er at give en fuldstændig klassifikation, men at give en idé om det høje vidensniveau, der kræves for både at udføre og modangreb.

### Phishing

Phishing-angreb er meget almindelige og nemme at udføre. De består i at sende e-mails, der ser ud til at komme fra en pålidelig afsender, såsom en kollega eller en bank, med det formål at narre offeret til at udføre en bestemt type operation.

Angriberen forsøger at overtale offeret til at åbne en vedhæftet fil eller giver et link, der fører til en side, der ligner den oprindelige side, for eksempel en bankhjemmeside, men oprettet af angriberen, for derefter at bede offeret om at indtaste deres legitimationsoplysninger. For at optimere denne type angreb er der en hel række af teknikker, der er en del af den såkaldte social engineering, da computersystemets sårbarhed i dette tilfælde skal identificeres hos brugeren selv, som bliver bedraget og utilsigtet giver sine akkreditiver.

For at forsvare sig mod disse angreb er det nødvendigt med en omhyggelig inspektion af linket (det er næsten umuligt at skelne mellem lille L 'l', et stort I 'I' og cifferet en '1', eller at bemærke forskellen mellem . det og .lt i en URL). Desuden skal man være opmærksom på, at seriøse virksomheder aldrig lægger links i deres e-mail, der bringer til en login-side.

### Adgangskodeangreb

For at få adgangskoder fra et offer kan en hacker prøve social engineering, som vist i det foregående afsnit, eller starte et adgangskodeangreb. Adgangskoder gemmes normalt som hashes, og hashes er meget svære at invertere. Der er grundlæggende to tilfælde: den krypterede adgangskode er tilgængelig eller ukendt, og denne type angreb kan udføres på forskellige måder:

- **Brute force angreb:** i dette tilfælde vil programmet prøve alle mulige kombinationer af karakterer ud af et sæt;
- **Angreb med ordbog:** programmet vil teste en sekvens af almindeligt anvendte adgangskoder (ordbog);

- **Angreb med regnbuefiler:** en regnbuetabel er en forudberegnet tabel, der viser outputtet af hash-funktioner af adgangskode. Brug af en nøgleafledning, der anvender et salt, gør dette angreb umuligt.

Systemadministratorer bør teste adgangskodefilerne for brugerne af deres systemer på regelmæssig basis med programmer, der er i stand til at udføre adgangskodeangreb, både med brute force og ved hjælp af ordbog.

For at beskytte mod disse angreb bør et system kontrollere kvaliteten af adgangskoden, før brugeren tillader det at indstille den, hvilket kræver brug af et minimum antal tegn (f.eks. 10) fra forskellige sæt (store og små bogstaver, cifre, tegnsætningstegn). ). Hvis den indtastede adgangskode er forkert for nogle efterfølgende forsøg, er det passende at vedtage en kontolås- eller forsinkelsespolitik ved næste forsøg, hvilket forhindrer angriberen i hurtigt at prøve mange kombinationer.

**Bufferoverløb**

Bufferoverløb (eller overløb) opstår, når du sender flere bytes til en buffer, end den kan rumme, disse bytes flyder over bufferen, der er beregnet til at holde dem og ændre andre bytes eller, afhængigt af operativsystemet og filtypen, endda injicere eksekverbar maskinkode. Mens moderne operativsystemer kan forhindre at køre en eksekverbar fil med blandet data og kode, er dette ikke sandt for de mere simple operativsystemer, der ofte findes i indlejrede systemer, hvor der i mange tilfælde slet ikke er noget operativsystem.

**SQL-injektion**

Et meget almindeligt angreb er SQL Injection. SQL-sproget bruges i relationelle databaser; derfor er disse typer angreb rettet mod informationstyveri. Grundet den fortolkede karakter af SQL-forespørgslen sendes ondsindet SQL-kode til en webside, der er kendt for at bruge en SQL-forespørgsel til at indsamle de data, der skal præsenteres for brugeren. Den ondsindede SQL-kode ændrer (nogle gange blot tilføjer andre syntaktiske dele) den foruddefinerede SQL-forespørgsel, der resulterer, er en modificeret forespørgsel, der afslører andre data end dem, den oprindelige forespørgsel blev skrevet til.

Selvom det ikke er komplekst at forhindre denne type angreb (for eksempel at forhindre adgang til andre tabeller i databasen ud over dem, der er nødvendige for at udføre den oprindelige forespørgsel), er SQL-injektion stadig et af de mest almindelige angreb.

**Cross-site Scripting (XSS)**

I en cross-site scripting (XSS) injiceres ondsindede eksekverbare scripts i koden for en betroet webapplikation (plugin) eller hjemmeside, med det resultat, at den downloades af andre personer

(seere). Når serveren sender den kompromitterede webside tilbage, anser brugerens webbrowser, at den er leveret fra en pålidelig kilde, og derefter identificeres den indsprøjtede kode ikke som værende ondsindet. Det aktive indhold er normalt et script (programmeringskode, der udføres af browseren), der kan tillade en hacker at få adgang til følsomt sideindhold, sessionscookies eller andre oplysninger, som browseren opbevarer på vegne af brugeren. Med XSS-angreb er det muligt at stjæle følsomme data fra alle brugere, der tilgår inficerede websteder. For eksempel kan stjæle SESSIONID'et for en webstedssession (som er placeret i brugerens browsercookies, der igen bruges til at undgå, at hver forbindelse til det samme websted kræver godkendelse ud over den første), give en bedrager mulighed for at omgå loginfasen og ulovligt operere på vegne af brugeren.

**Cross-Site Request Forgery (CSRF)**

En CSRF er en ufrivillig http-anmodning (ønsket af angriberen) til et websted udført af en bruger, der allerede er godkendt til det pågældende websted; hjemmesiden vil på dette tidspunkt udføre anmodningen uden yderligere verifikation, hvilket giver angriberen mulighed for at udføre sit angreb. Denne type angreb gør det muligt at ændre brugerdata eller udføre uønskede transaktioner, såsom uønskede køb, hvis det for eksempel var en e-handelsside. Angriberen skal have meget godt kendskab til det pågældende sted for at udføre et CSRF-angreb.

Et eksempel på et CSRF-angreb udføres ved at ændre en URL. For eksempel analyseres kildekoden på websiden af malwaren for at finde "Skift adgangskode"-linket og verificere, at http GET-metoden er brugt, så kan der dannes et nyt link med en ny adgangskode (valgt af angriberen) og sendt til offeret, for eksempel via en vedhæftet fil i e-mail. Hvis der klikkes på linket, og det sker, at offeret allerede er godkendt til webstedet, aktiveres en skjult start af adgangskodeændringsprocedure, hvilket effektivt giver kontoen til hackeren. Gode metoder til ændring af adgangskode beder om at indsætte den gamle adgangskode også (og GET-metoden bruges ikke). Desuden er brugen af en CAPTCHA-metode (en test til at kontrollere, om brugeren er et menneske eller en computer) ganske effektiv.

**Filrelaterede angreb**

Der er mange typer angreb på filer. De mest kendte sårbarheder er:

- **Usikre direkte objektreferencer:** opstår, når en applikation anmoder om en ressource fra en server, der kalder den ved sit navn, men tillader brugeren at ændre sidstnævnte for at anmode om en anden ressource.
- **Local File Inclusion (LFI):** udnytter sårbarheden af GET- og POST-funktionerne i PHP-sproget til at indsætte ondsindede filer i koden.
- **Path Traversal:** Tillader en ondsindet bruger at få adgang til mapper med begrænset adgang og udføre kommandoer. Ofte bruges de efter en LFI til at komme dybt ind i systemet og opnå de nødvendige privilegier.

For at beskytte mod disse typer angreb er det vigtigt at forhindre filupload til servere, deaktivere fjernudførelse af kode, begrænse PHP-adgang til filsystemet og opdatere software med jævne mellemrum.

**Denial of Service (DoS) og Distributed Denial of Service (DDoS)**

Måske de mest kendte netværksangreb, DoS-angreb består grundlæggende i at sende mange pakker for at blokere muligheden for at modtage andre pakker, og dermed forhindre adgang til maskinen for almindelige brugere, blokere de tilbudte tjenester.

Forskellen på et DoS- og et DDoS-angreb ligger i, at i det første tilfælde bruges en enkelt vært til at udføre angrebet, mens angrebet i det andet udføres med mange værter (kaldet zombier) på samme tid. Disse værter er tidligere blevet inficeret af en malware og på kommando af angriberen starter forbindelsen til offermaskinen.

Blandt DoS-angrebene kan vi huske:

- **Syn-Flood** : består af at sende en stor mængde forbindelsesanmodninger. Maskinen, der modtager en anmodning (en SYN-pakke) fra en klient, vil svare klienten ved at sende en besked, allokere nogle ressourcer til denne forbindelse og vente på bekræftelsespakken (der aldrig kommer). Derfor forbliver forbindelsesanmodningen aktiv, indtil der opstår en timeout. Hvis alle mulige tilslutninger er startet, vil den service, der leveres af maskinen, ikke længere være tilgængelig. Timeoutet indtræffer inden for få sekunder, men den enorme hastighed af anmodninger holder maskinen tilstoppet.
- **Smurf** : dette angreb er en kombination af IP-spoofing og ICMP-anmodninger. IP-spoofing refererer til afsendelsen af en IP-pakke, hvor afsenderens IP er forfalsket. Ved kontinuerligt at sende *ICMP ekko-anmodninger* (meddelelsen sendt af ping-kommandoen, bruges til at bekræfte, om en maskine er tilgængelig) for at udsende IP-adresser efter at have erstattet afsenderens IP-adresse med offerets IP-adresse, vil offermaskinen modtage et svar for hver anmodning fra enhver vært aktiv på netværket og blokerer dermed forbindelsen (oversvømmelse). For at forhindre, at disse angreb opstår, er det muligt at deaktivere videresendelse af udsendelses-IP-adresser i routerne.
- **Ping of death** : består i at sende en fragmenteret pakke med en samlet IP-størrelse, der er større end den maksimalt tilladte til en offermaskine, når først maskinen samler pakken igen, kan den medføre bufferoverløb.

Det er almindeligt for et DDoS-angreb at bruge et **Botnet** , der er tusinder eller millioner af computere inficeret med malware, som på kommando af angriberen udfører et Syn-Flood-angreb i forening på en offerserver.

## Man in The Middle (MITM)

Med MITM mener vi en hel typologi af angreb, der har til formål at opsnappe, analysere og genudsende en kommunikation mellem to parter, der mener, at de kommunikerer direkte til hinanden.

Denne type angreb udføres normalt ved hjælp af IP-spoofing: netværkssniffere aflurer en forbindelse og nye pakker oprettes ved at ændre IP-adressen på afsenderen og modtageren, så afsenderen sender data til angriberen, som igen sender dem til modtageren og omvendt.

Der er mange måder at forsvare sig mod disse typer angreb på; den mest effektive og brugte i dag er at indføre *end-to-end kryptering* , hvor slutpersonerne udfører krypteringen/dekrypteringen og alle mellemsystemerne kun kan se krypterede data.

## ARP-spoofing eller ARP-forgiftning

Address Resolution Protocol (ARP) gør det muligt for netværkskommunikation at nå en bestemt enhed på det lokale netværk. Det bruges til at kortlægge IP-adresser til Media Access Control (MAC)-adresser (bruges til at identificere værterne på et LAN). En vært skal bruge ARP-protokollen for at kende MAC-adressen på routeren/gatewayen for at oprette forbindelse til internettet. Hver vært opretholder en *ARP-cache* for at kortlægge allerede anmodede og kendte IP/MAC-par. Hvis værten ikke kender MAC-adressen for en bestemt lokal IP-adresse, sender den en broadcast ARP-anmodningspakke og spørger alle de andre værter på det lokale netværk, hvem der har IP-adressen. Værten med denne IP-adresse vil svare og angive sin MAC-adresse.

Et ARP-spoofing/forgiftningsangreb kan klassificeres som et MITM-angreb på lokalt netværk. Angriberen skal have adgang til det lokale netværk og scanner det for at bestemme IP-adresserne på routeren og en anden vært. Angriberen sender forfalskede ARP-svar, der får routeren og værten til at tro, at angriberens MAC-adresse er henholdsvis værtens og routerens ene. Dette narre både router og arbejdsstation til at oprette forbindelse til angriberens maskine i stedet for til hinanden. Når de to enheder opdaterer deres ARP-cache-poster, kommunikerer vært og router med angriberen i stedet for direkte med hinanden.

## DNS-spoofing

Domain Name System (DNS) er det navngivningssystem, der bruges til at konvertere internetnavne (mere korrekt FQDN - Fully Qualified Distinguished Names, f.eks. www.server.net) til den tilsvarende IP-adresse. Systemet er hierarkisk og decentraliseret med nogle servere på øverste niveau. Hver gang et system forsøger at nå en netværksressource, oprettes en forbindelse til en

DNS-server for at konvertere navnet på ressourcen til en IP-adresse. Ressourceposterne indeholder normalt administrative oplysninger og andre data, som f.eks. bruges af mailservere.

En hacker vil gerne ændre DNS-poster for at sende trafik til et falsk eller "forfalsket" websted i stedet for det legitime. Den spoofede maskine kunne derefter udføre MITM-angreb, spoofing osv. For at forhindre DNS-spoofing skal DNS-servere holdes opdaterede.

**Drive-by-angreb**

I et drive-by-angreb er ondsindet kode indlejret i et usikkert websted. Når den inficerede side besøges, udføres scriptet automatisk på offerets computer og kompromitterer det, der er ingen grund til at gøre noget på webstedet eller indtaste nogen information. I dette tilfælde er brugen af en opdateret nyere og anstændig antivirus og/eller personlig firewall den eneste beskyttelse, da de kan registrere fra deres database, hvis et websted er usikkert, før en bruger besøger det.

**Genbrug af legitimationsoplysninger**

Behovet for at have legitimationsoplysninger til flere tjenester får brugerne til at genbruge nogle af deres adgangskoder. Når først en angriber har en liste over brugernavne og adgangskoder til kompromitterede websteder eller tjenester (f.eks. hentet fra Dark Web), er det muligt, at den samme bruger kan bruge det samme bruger/adgangskodepar på andre systemer. Dette vil i høj grad reducere antallet af adgangskoder til at kontrollere med et ordbogsangreb. For at modsætte denne type angreb er de samme modforanstaltninger, der er vedtaget for at imødegå adgangskodeangreb, effektive. Når der er brug for mange komplekse adgangskoder, finder nogle mennesker adgangskodeadministratorer nyttige.

**Websessionkapring**

Et Session Hijacking-angreb består af udnyttelse af sessionstokens, der bruges af webbrowseren som sessionskontrolmekanismer. Webserveren skal relatere forskellige forbindelser fra den samme bruger, da der for hver webside oprettes en TCP-forbindelse for hvert element (tekst, billeder osv.). Den mest anvendte metode er baseret på et token, som webserveren sender til klientbrowseren efter en vellykket klientgodkendelse. Et sessionstoken er blot en identifikator, en lang unik streng, og kan bruges til at undgå behovet for at udføre en godkendelsesprocedure for hver forbindelse. Dette angreb udføres ved at stjæle eller forudsige et gyldigt sessionstoken for at få uautoriseret adgang til webserveren.

**TCP-sessionkapring**

TCP-forbindelser skal garantere at levere pakker i den rigtige sekventielle rækkefølge, for at opnå dette bruger den bekræftelsespakker og sekvensnumre. Disse bekræftelsespakker er blot tal, der

på en eller anden måde øges efter det 3-vejs indledende håndtryk. Målet med TCP session hijacker er at skabe en tilstand, hvor klienten og serveren ikke er i stand til at udveksle data; sætter ham/hende i stand til at smede acceptable pakker til begge ender, som efterligner de rigtige pakker. Således er angriberen i stand til at få kontrol over sessionen. Hvis sekvensen kan forudsiges, kan en angriber sende forfalskede pakker, der genkendes som gyldige fra offerværten.

**IP-spoofing**

En IP-spoofing udføres normalt med et DoS-angreb: et stort antal forbindelser blokerer den legitime server, og på samme tid sætter en falsk server sin IP-adresse med nummeret på den legitime server. Derfor bliver brugeren kørt til den falske server, der sender beskeder med en IP-adresse, der indikerer, at beskeden kommer fra en betroet vært.

Blind hijacking: I tilfælde, hvor kilderouting er deaktiveret, kan sessionskaperen også bruge blind kapring, hvor han injicerer sine ondsindede data i opsnappet kommunikation i TCP-sessionen. Det kaldes blind, fordi han ikke kan se svaret; Selvom flykapreren kan sende data eller kommandoer, gætter han grundlæggende på klientens og serverens svar.

**Nul-dages udnyttelse**

Strengt taget er zero-day exploit ikke et specifikt angreb, da cyberkriminelle får kendskab til en sårbarhed, der netop er blevet opdaget i visse software- og operativsystemer, og derefter retter sig mod organisationer, der bruger denne software, før en rettelse bliver tilgængelig.

**Læk af kode**

Kildekoden til applikationer kan nogle gange indeholde følsomme oplysninger og give værdifuld indsigt for et cyberangreb, ud over den iboende økonomiske værdi af selve koden.

## Spørgsmål :

I separat fil.

## Vedhæftede filer :

I separat fil.

## Referencer:

Hvert emne har sin egen side på Wikipedia, nogle gange allerede for grundig til formålet med dette kursus; bøger er for specialiserede og er kun nyttige efter en meget dyb træning om det specifikke emne.

1. Malware eller skadelig software
   URL: https://www.malwarebytes.com/malware
2. Cybersikkerhed for alle.
   https://www.avast.com/c-malware
3. Hvad er cybersikkerhedstrusler?
   URL: https://www.imperva.com/learn/application-security/cyber-security-threats/
4. 21 Top cybersikkerhedstrusler og hvordan trusselsintelligens kan hjælpe
   URL: https://www.exabeam.com/information-security/cyber-security-threat/