Mindmap 001 — De CIA-triade

1. Wat is de CIA-triade?

Drie doelen van **informatiebeveiliging**:

- **Vertrouwelijkheid** — alleen bevoegden hebben toegang.
- **Integriteit** — informatie blijft correct en volledig.
- **Beschikbaarheid** — bevoegden kunnen informatie en diensten gebruiken wanneer nodig.

2. Vertrouwelijkheid

Bescherm informatie tegen inzage door onbevoegden.

- Beheer wie toegang krijgt en geef alleen rechten die nodig zijn voor de taak.
- Sterke authenticatie en versleuteling kunnen vertrouwelijkheid ondersteunen.

3. Integriteit

Houd informatie correct en **volledig**; voorkom ongeoorloofde of onbedoelde wijzigingen.

- Wijzigingsbeheer, beperkte schrijfrechten en controles kunnen hierbij helpen.
- Back-ups ondersteunen mogelijk herstel, maar voorkomen op zichzelf geen wijzigingen.

4. Beschikbaarheid

Zorg dat bevoegde gebruikers informatie en diensten kunnen gebruiken wanneer dat nodig is.

- Storingen, apparateuruitval en overbelasting kunnen beschikbaarheid aantasten.
- Redundantie, back-ups, herstelprocedures en capaciteitsbewaking kunnen beschikbaarheid ondersteunen.
- Test herstelprocedures.

5. Bedreigingen en voorbeelden

Een incident kan meer dan één doel van de CIA-triade raken.

- Gestolen inloggegevens kunnen leiden tot onbevoegde inzage.
- Een onbedoeld gewijzigd rekeningnummer kan een betalingsproces verstoren.
- Een storing of aanval die diensten overbelast, kan een systeem tijdelijk onbereikbaar maken.

6. Mitigerende maatregelen

Kies maatregelen die risico's verminderen en passen bij de informatie en werkwijze.

- **Vertrouwelijkheid:** toegangsbeheer, sterke authenticatie en versleuteling.
- **Integriteit:** wijzigingsbeheer, beperkte schrijfrechten en controles.
- **Beschikbaarheid:** redundantie, back-ups, geteste herstelprocedures en capaciteitsbewaking.

7. Afwegingen in de praktijk

Maatregelen hebben voordelen én beheerlasten. Geen enkele maatregel voorkomt elk risico.

- Versleuteling ondersteunt vertrouwelijkheid; zorgvuldig sleutelbeheer blijft belangrijk.
- Toegangscontroles beschermen informatie, maar mogen noodzakelijk werk niet onnodig blokkeren.
- Stel **prioriteiten af** op de informatie, gebruikers en werkwijze. Maatregelen verminderen risico's, maar garanderen niet dat incidenten uitblijven.

8. Korte samenvatting

Bescherm drie doelen: vertrouwelijkheid, integriteit en beschikbaarheid. Kies passende maatregelen en houd rekening met hun beperkingen.

- **Vertrouwelijkheid:** toegang voor bevoegden.
- **Integriteit:** informatie correct en volledig houden.
- **Beschikbaarheid:** diensten bruikbaar houden wanneer nodig.

Risicomanagement & Dreigingsmodellering

1. Wat is risicomanagement?

- Herkennen en beoordelen van gebeurtenissen die van invloed kunnen zijn op doelen, informatie en diensten.
- Risico verbindt een gebeurtenis met de mogelijke gevolgen.
- Breng activa, dreigingen, kwetsbaarheden en gevolgen in kaart.
- Evalueer en herzie regelmatig, bijvoorbeeld bij wijzigingen in systemen, processen of de context.

2. Inventariseer risico's

- Bepaal wat je wilt **beschermen**, wat er mis kan gaan en waarom.
- **Activa:** bijvoorbeeld account, databestand, dienst.
- **Dreiging:** bijvoorbeeld misbruik van een account of uitval van een dienst.
- **Kwetsbaarheid:** een zwakte die kan worden uitgebuit.
- **Gevolgen:** bijvoorbeeld ongeoorloofde inzage, verkeerde gegevens of onderbreking van een dienst.

3. Schat waarschijnlijkheid en impact

- Gebruik een kwalitatieve waarschijnlijkheid-impactmatrix om te prioriteren, niet als exacte voorspelling.
- Schat de waarschijnlijkheid dat een dreiging zich voordoet.
- Schat de impact als het gebeurt.
- Gebruik lage, middelhoge en hoge niveaus (laag / middel / hoog).
- Leg aannames vast en houd rekening met de context.

4. Classificeer dreigingen met STRIDE

- **Identiteitsvervalsing:** kan iemand zich voordoen als een ander?
- **Manipulatie:** kunnen gegevens of functionaliteit worden gewijzigd?
- **Ontkenning:** kan iemand handelingen ontkennen en is er voldoende bewijs?
- **Openbaarmaking van informatie:** kan gevoelige informatie worden ingezien?
- **Denial-of-service (dienstverstoring):** kan de beschikbaarheid worden verstoord?
- **Verhoging van bevoegdheden:** kan iemand meer rechten krijgen dan bedoeld?

5. Kies een behandelstrategie

- **Vermijden:** stop of verander de activiteit die het risico veroorzaakt.
- **Verminderen:** neem maatregelen om de waarschijnlijkheid en/of impact te beperken.
- **Overdragen:** leg het risico (deels) bij een ander, bijvoorbeeld via afspraken of een verzekering; het risico blijft bestaan.
- **Accepteren:** accepteer het resterende risico bewust en leg dit vast.

6. Voorbeeld: onbeveiligde beheerdersaccount

- Een onbeveiligde beheerdersaccount kan leiden tot ongeoorloofde toegang tot systemen.
- **Activa:** account, systemen.
- **Dreigingen:** identiteitsvervalsing of verhoging van bevoegdheden.
- **Gevolgen:** inzage in gegevens, wijzigingen aan systemen, verstoring van diensten.
- De inschatting van waarschijnlijkheid en impact is afhankelijk van de context en is kwalitatief, geen zekerheid.
- **Maatregelen:** pas least privilege toe en gebruik sterkere authenticatie (bijvoorbeeld multi-factor authenticatie).

7. Leg de afweging vast

- Leg aannames en mogelijke gevolgen vast.
- Koppel de gekozen maatregel aan de dreiging en het betreffende actief.
- Leg het resterende risico en de beslissingnemer vast.
- Herbeoordeel wanneer de omstandigheden veranderen of de effectiviteit van maatregelen wijzigt.

8. Korte samenvatting

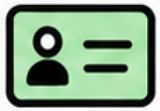
- Breng activa, dreigingen, kwetsbaarheden en gevolgen in kaart.
- Gebruik een kwalitatieve inschatting van waarschijnlijkheid en impact.
- Classificeer dreigingen met de zes STRIDE-categorieën.
- Kies uit vier behandelstrategieën.
- Risicomanagement verkleint risico's, maar biedt geen garantie dat incidenten volledig verdwijnen.

Grondbeginselen van authenticatie en autorisatie

1. Wat is het?

- Authenticatie controleert wie of wat je bent.
- Autorisatie bepaalt wat je daarna mag doen.

2. Identiteitsverificatie



- Een gebruiker, apparaat of ander onderwerp bewijst een opgegeven identiteit.
- Een dienst controleert die aanwijzingen voordat toegang wordt verleend.

3. Authenticatiefactoren

- **Kennis**: iets wat je weet, zoals een wachtwoord.
- **Bezit**: iets wat je hebt, zoals een beveiligingssleutel.
- **Biometrie**: iets wat je bent, zoals een vingerafdruk.
- Multifactorauthenticatie vereist factoren uit verschillende categorieën. Meerdere stappen uit dezelfde categorie tellen niet als verschillende factorsoorten; twee wachtwoorden zijn bijvoorbeeld allebei kennis.

4. Authenticatie in de praktijk

- Je meldt je aan met een wachtwoord en eventueel een tweede factor.
- Een factor of biometrisch kenmerk is niet op zichzelf foutloos.
- Volg de beveiligingsinstructies van de dienst.

5. Autorisatie

- Na authenticatie controleert het systeem welke toegang is toegestaan.
- Voorbeeld: een account mag een document lezen, maar niet wijzigen.

6. RBAC en ABAC

- **RBAC** koppelt machtigingen aan **rollen**; gebruikers krijgen toegang via hun rol.
- Voorbeeld: de rol ‘lezer’ verleent leesrechten.
- **ABAC** beoordeelt kenmerken van gebruiker, bron, actie of context.

7. Minimale bevoegdheden

- Geef een gebruiker of proces alleen de rechten die nodig zijn voor de taak.
- Verleen niet meer toegang dan noodzakelijk.

8. Korte samenvatting

- Authenticatie: wie of wat ben je?
- Autorisatie: wat mag je doen?
- Gebruik passende factoren, rollen of kenmerken en beperk toegangsrechten tot wat nodig is.

Grondbeginselen van logboekregistratie en auditing

Logs helpen gebeurtenissen te volgen en te onderzoeken, maar bewijzen op zichzelf niet dat een gebeurtenis zeker heeft plaatsgevonden.

1. Wat is het

- Een beveiligingslogboek registreert gebeurtenissen die van belang zijn voor de beveiliging van een systeem of netwerk.
- Voorbeelden: aanmeldingen, geweigerde toegang, configuratiewijzigingen en waarschuwingen.
- Een audittrail brengt activiteiten en wijzigingen samen, zodat gebeurtenissen achteraf te volgen zijn.

2. Waarom loggen

- Logboeken helpen gebeurtenissen te begrijpen en mogelijke beveiligings-problemen te onderzoeken.
- Breng gebeurtenissens in volgorde en leg verbanden tussen systemen.
- Gebruik meldingen als aanleiding voor onderzoek; een logregel alleen bewijst niet dat iets zeker is gebeurd.

3. Wat registreer je

- Leg genoeg context vast om een gebeurtenis te kunnen begrijpen, zonder onnodige gegevens te verzamelen.
- Tijdstip, bron, account of systeem, gebeurtenistype en resultaat.
- Bijvorobeeld: aanmelding mislukt; account-ID; systeem; tijdstip; resultaat.
- Tijdssynchronisatie helpt gebeurtenissen te ordenen. Een tijdstempel bewijst niet dat de klok nauwkeurig was.

4. Traceerbaarheid en audittrail

- Traceerbaarheid koppelt gebeurtenissen aan relevante accounts, systemen en tijdstippen.
- Unieke accountnamen en consistente identificatie ondersteunen die koppeling.
- Gedeelde accounts maken het moeilijker om een handeling aan één persoon toe te schrijven.
- Een audittrail maakt achteraf zichtbaar wat er gebeurde, wanneer en via welk account of systeem.

5. Integriteit en niet-weerlegbaarheid

- Niet-weerlegbaarheid houdt in dat iemand een handeling niet geloofwaardig kan ontkennen op basis van de beschikbare bewijsketen.
- Logging alleen garandeert dit niet.
- Beperk inzage, wijziging en verwijdering; betrouwbare identificatie en integriteits-bescherming zijn ook van belang.
- Een tijdstempel of logregel op zichzelf bewijst niet dat een gebeurtenis daadwerkelijk plaatsvond.

6. Bewaren en beschermen

- Bescherm logs tegen ongeautoriseerde inzage, wijziging en verwijdering. Bepaal een passende bewaartermijn voor het doel en de context.
- Beperk schrijftoegang en scheid beheertaken waar dat passend is.
- Opslag buiten het systeem dat de logs maakt, kan de integriteit versterken.
- Bewaartermijnen hangen af van doel, afspraken, toepasselijke regels en opslagruimte; er is geen universele termijn.
- Verwijder gegevens na afloop van de vastgestelde termijn gecontroleerd.

Let op gevoelige gegevens — Logs kunnen persoonsgegevens of geheimen bevatten. Beperk toegang en leg niet meer vast dan nodig; behandel logs volgens het beveiligings- en privacybeleid.

7. Praktijkvoorbeeld

- Een aanmelding mislukt. De registratie bevat de noodzakelijke context en wordt alleen toegankelijk gemaakt voor bevoegde personen.
- Leg tijdstip, account-ID, systeem en resultaat vast.
- Koppel een relevante melding aan een onderzoek.
- Voeg geen onnodige persoonsgegevens of geheimen toe.

8. Korte samenvatting

- Goede logging maakt gebeurtenissen beter te volgen en te onderzoeken, maar is op zichzelf geen sluitend bewijs.
- Leg relevante gebeurtenissen met voldoende context vast.
- Gebruik unieke identificatie en ondersteun de integriteit van logs.
- Bescherm logs, beperk gevoelige gegevens en stel een passende bewaartermijn vast.

005 · Beveiligingsbeleid en normenkaders

1. Wat is het?

Beveiligingsbeleid legt vast hoe een organisatie informatie en systemen beschermt.

- Geeft richting aan keuzes en dagelijks handelen.
- Verbindt risico's, doelen en passende maatregelen.

2. Beleidsarchitectuur

Werk van algemene afspraken naar uitvoerbare instructies.

- Overkoepelend informatiebeveiligingsbeleid: doelen, reikwijdte en uitgangspunten.
- Onderliggende beleidsregels en standaarden: concrete eisen en normen.
- Procedures en werkinstructies: wie doet wat, wanneer en hoe.
- Wijs een eigenaar aan, leg goedkeuring vast en herzie periodiek.

3. Risico's en beheermaatregelen

Een risicobeoordeling helpt bepalen welke maatregelen passen bij de organisatie en haar risico's.

- Organisatorisch: incidentprocedure en duidelijke afspraken.
- Personeel: bewustwording en training.
- Technisch: toegangsbeheer, logging en back-ups.
- Leg keuzes en uitzonderingen vast; evalueer of maatregelen werken.

4. Rollen en verantwoordelijkheden

Beleid werkt wanneer duidelijk is wie besluiten neemt, uitvoert en controleert.

- Wijs een eigenaar aan voor beleid en periodieke herziening.
- Beschrijf verantwoordelijkheden voor goedkeuring, uitvoering en opvolging.
- Stem rollen af op de omvang, context en risico's van de organisatie.

5. Naleving en documentatie

Maak afspraken controleerbaar: wijs verantwoordelijkheden toe, bewaar bewijs en controleer periodiek.

- Identificeer relevante interne beleidsregels en toepasselijke verplichtingen.
- Bewaar bijvoorbeeld goedgekeurd beleid, trainingsregistraties en evaluaties.
- Controleer of afspraken worden nageleefd en leg verbeteracties vast.

Let op: Dit is een algemeen governanceprincipe, geen juridisch advies of garantie op naleving van alle verplichtingen.

6. ISO/IEC 27001 en ISMS

ISO/IEC 27001 beschrijft eisen voor een managementsysteem voor informatiebeveiliging (ISMS; ook SGSI genoemd).

- Het ISMS wordt opgezet, uitgevoerd, onderhouden en voortdurend verbeterd.
- Het normenkader is niet hetzelfde als het beleid: het kan helpen beleid en maatregelen te structureren.
- Stem de inrichting af op risico's en organisatiecontext.

7. NIST CSF: vijf functies

Het NIST Cybersecurity Framework ordent cybersecurity-activitieten in vijf samenhangende functies; het is geen verplichte lineaire volgorde.

- **Identificeren (Identify)**: breng organisatie, systemen, gegevens, risico's en verantwoordelijkheden in beeld.
- **Beschremen (Protect)**: pas waarborgen toe, zoals toegangsbeheer, training en andere maatregelen.
- **Detecteren (Detect)**: signaleer en analyseer gebeurtenissen tijdig.
- **Reageren (Respond)**: coördineer communicatie en beheers incidenten.
- **Herstellen (Recover)**: herstel diensten en verwerk leerpunten.

8. Korte samenvatting

Gebruik beleid en normenkaders samen om beveiliging doelgericht in te richten en te verbeteren.

- ISO/IEC 27001 beschrijft eisen voor een ISMS; NIST CSF ordent activitieten in vijf functies.
- Voorbeeld: inventariseer een kritieke dienst en risico's; beperk toegang en maak back-ups; bewaak meldingen; volg bij een alarm de incidentprocedure; herstel de dienst en verwerk leerpunten.

Menselijke factor & security awareness

Herken signalen • verifieer veilig • meld tijdig

1 Wat is security awareness

- Risico's herkennen
- Veilig dagelijks gedrag
- Vragen stellen en verdachte gebeurtenissen melden

2 Waarom het belangrijk is

- Manipulatie kan informatie prijsgeven of handelingen afdwingen
- Alertheid en vroeg melden helpen risico's beperken
- Eén afwijkend signaal bewijst geen aanval

3 Manipulatietechnieken

- Phishing via e-mail of berichten
- Zich voordoen als collega of leverancier
- Urgentie veinzen
- Vertrouwen misbruiken of vertrouwelijke gegevens vragen

4 Herkennen

- Onverwacht verzoek of ongebruikelijke afzender
- Verdachte of onverwachte link
- Tijdsdruk
- Vraag om wachtwoord of verificatiecode
- Afwijking van normale procedures
- Context telt: één signaal bewijst geen aanval

5 Training

- Korte, herhaalde leermomenten
- Herkenbare scenario's bespreken
- Veilig melden oefenen
- Afstemmen op rol en werkomgeving
- Zonder beschuldigingen

6 Veilig reageren

- Pauzeer; klik niet op verdachte links
- Deel geen wachtwoorden of verificatiecodes
- Controleer via bekend, onafhankelijk kanaal

Tip: gebruik geen contactgegevens of links uit het verdachte bericht voor controle.

7 Verdachte situaties melden

- Verspreid bericht of voorval niet verder
- Gebruik aangewezen meldkanaal
- Geef tijdstip en afzender door
- Volg instructies beveiligingsteam
- Meld mogelijke vergissing snel
- Verander of wis geen bewijsmateriaal tenzij de organisatie daarom vraagt

8 Korte samenvatting

- Herken
- Verifieer onafhankelijk
- Meld tijdig

Mindmap 007: Social engineering

1. Wat is social engineering?

- Misleiding of sociale druk om iemand informatie te laten delen, een handeling te laten uitvoeren of toegang te verlenen.

2. Phishing en spear-phishing

- **Phishing:** misleidend bericht dat bijvoorbeeld naar een valse inlogpagina leidt.
- **Spear-phishing:** gericht op een specifieke persoon of organisatie; kan persoonlijke context gebruiken.
- **Voorbeeld:** onverwacht verzoek om via een link je wachtwoord opnieuw in te stellen.

3. Vishing en smishing

- **Vishing:** misleiding via spraak of telefoongesprek.
- **Smishing:** misleidend sms- of ander tekstbericht op een mobiel apparaat.
- De afzender kan zich voordoen als een vertrouwde organisatie.

4. Pretexting en baiting

- **Pretexting:** een verzonnen rol of situatie aannemen om vertrouwen te winnen of informatie te krijgen.
- **Baiting:** lokken met een aantrekkelijk aanbod of object, zoals een onbekende USB-stick.
- Deel geen gegevens en sluit geen onbekende apparaten aan.

5. Tailgating en fysieke toegang

- **Tailgating:** zonder toestemming meelopen achter iemand die toegang heeft tot een beveiligde ruimte.
- Volg de toegangsprocedure; laat onbekenden niet op jouw pas binnen.
- Vraag bij twijfel hulp aan beveiliging of een verantwoordelijke medewerker.

6. Herkenningssignalen

- Onverwacht verzoek om een wachtwoord, betaling, bestand of toegang.
- Druk om snel te handelen of een verzoek geheim te houden.
- Ongebruikelijke link, afzender, situatie of manier van contact.

7. Verifiëren, voorkomen en melden

- Controleer ongebruikelijke verzoeken via een bekend, onafhankelijk contactpunt.
- Gebruik minimale toegangsrechten en volg fysieke toegangscontroles.
- Meld verdachte berichten of situaties via het interne meldkanaal.
- Organisaties kunnen periodieke bewustwordingsoefeningen en duidelijke meldprocedures inzetten.

8. Korte samenvatting

- Herken de druk of misleiding; handel niet overhaast.
- Gebruik geen verdachte link. Open zelf de bekende officiële website of meld het bericht volgens de interne procedure.
- Verifieer via een bekend contactpunt en meld twijfel.

Fysieke beveiliging van IT-infrastructuren

1. Wat is fysieke beveiliging?

- Bescherm mensen, apparatuur en informatie tegen fysieke schade en onbevoegde toegang.
- Combineer passende maatregelen: geen enkele maatregel volstaat op zichzelf.
- Voorbeeld: serverruimte heeft gecontroleerde toegang én bescherming tegen omgevingsrisico's.

2. Perimeter

- Beveilig de fysieke grens van een locatie en toegang tot gebouwen.
- Afsluitingen en passende terreinverlichting kunnen onbevoegde toegang ontmoedigen of signaleren.
- Gecontroleerde toegang tot gebouwen vormt een aanvullende laag.

3. Toegangscontrole

- Bepaal wie toegang krijgt tot een ruimte of voorziening.
- Identificeer personen en autoriseer toegang, bijvoorbeeld met badges of sleutels.
- Beperk gevoelige ruimten tot bevoegde personen.
- Registratie of toezicht kan helpen gebeurtenissen achteraf vast te stellen.

4. Bezoekers en zones

- Maak toegang voor bezoekers beheersbaar en stem die af op de ruimte.
- Registreer bezoekers volgens het beleid van de locatie.
- Geef bezoekers alleen toegang tot de zones die zij mogen betreden.
- Begeleiding of toezicht kan helpen om toegestane routes te volgen.

5. Koeling en omgevingsrisico's

- Bescherm apparatuur tegen omstandigheden die werking verstoren of schade veroorzaken.
- Koeling helpt apparatuur binnen geschikte bedrijfstemperaturen te houden.
- Houd rekening met warmte, rook, brand en waterschade.
- Keuze en ontwerp hangen af van locatie, apparatuur en toepasselijke eisen.

6. Branddetectie en blussing

- Detectie waarschuwt voor brand; een blussysteem is bedoeld om brand te bestrijden.
- Branddetectie en blussing hebben verschillende functies.
- Passende oplossing hangt af van locatie en aanwezige apparatuur.
- Ontwerp en toepassing houden rekening met relevante eisen.

7. Bekabeling en clean desk

- Bescherm bekabeling en voorkom onbeheerde gevoelige informatie.
- Kabelgoten, afgesloten ruimten en ordelijke routing beschermen kabels tegen schade en onbevoegde toegang.
- Redundante verbindingen ondersteunen weerbaarheid, maar garanderen geen beschikbaarheid.
- Berg gevoelige documenten en verwijderbare opslagmedia op wanneer ze niet worden gebruikt.
- Vergrendel het scherm wanneer je de werkplek verlaat.

8. Korte samenvatting

- Fysieke beveiliging werkt het best als samenhangende lagen.
- Beveilig perimeter en beperk toegang tot bevoegde personen.
- Bescherm apparatuur tegen omgevingsrisico's en let op kabelroutes.
- Laat de werkplek veilig achter: berg gevoelige spullen op en vergrendel het scherm.

DNS-architectuur en DNS-beveiliging

1. Wat is het

- DNS vertaalt domeinnamen naar gegevens, zoals IP-adressen.
- Domeinnaam: bijvoorbeeld voorbeeld.nl
- Antwoord: bijbehorend adres of andere DNS-gegevens
- Een naam gebruiken in plaats van een numeriek adres

2. DNS-resolutie

- Een resolver zoekt het antwoord; soms uit cache, soms via andere DNS-servers.
- Voorbeeld: apparaat vraagt adres van voorbeeld.nl op
- Cache gevonden: opgeslagen antwoord terug
- Geen cacheantwoord: recursieve en iteratieve verzoeken

3. Cache poisoning

- Onterecht gewijzigde cache kan een domein naar een verkeerde bestemming sturen.
- Risico: onjuist DNS-antwoord
- Verdediging: betrouwbare DNS-diensten; onderzoek beveiligingswaarschuwingen
- DNSSEC-validatie controleert authenticiteit en integriteit

4. DNS-tunneling

- Communicatie of data verborgen in DNS-query's en -antwoorden kan controles omzeilen.
- Let op afwijkend DNS-verkeer
- Verdediging: monitor en onderzoek ongebruikelijke patronen
- Beperk DNS-verzoeken volgens netwerkbeleid

5. DNS-hijacking

- Manipulatie van DNS-resolutie kan gebruikers naar een ongewenste bestemming sturen.
- Mogelijk: gewijzigde instellingen of gemanipuleerde antwoorden
- Verdediging: controleer instellingen; onderzoek onverwachte omleidingen
- Passende netwerkbeveiliging; houd systemen up-to-date

6. DoH en DoT

- DoH (DNS over HTTPS) en DoT (DNS over TLS) versleutelen verkeer tussen client en resolver.
- Beschermen verkeer onderweg tegen meekijken
- Een resolver is daardoor niet automatisch betrouwbaar
- Voorkomen niet alle DNS-aanvallen

Onthoud — Versleuteling beschermt transport; vertrouwenscontrole blijft nodig.

7. DNSSEC implementeren

- DNSSEC controleert authenticiteit en integriteit, maar versleutelt DNS-verkeer niet.
- Conceptueel: onderteken de DNS-zone
- Publiceer sleutels en handtekeningen
- Bovenliggende zone publiceert DS-record
- Resolvers voeren validatie uit

Let op — Conceptueel overzicht, geen configuratiehandleiding.

8. Korte samenvatting

- DNS koppelt domeinnamen aan gegevens en maakt diensten vindbaar.
- Cache poisoning en hijacking kunnen naar verkeerde bestemming leiden
- Tunneling kan controles omzeilen
- DoH/DoT versleutelen onderweg; DNSSEC controleert authenticiteit en integriteit

Grondbeginselen van symmetrische cryptografie

1. Wat is het

- Eén geheime sleutel versleutelt én ontsleutelt gegevens.
- Afzender en ontvanger moeten die sleutel vertrouwelijk delen.

2. Blok- en stroomcijfers

- **Blokcijfer:** verwerkt gegevens in blokken van vaste lengte.
- **Stroomcijfer:** versleutelt gegevens als een stroom, doorgaans met een sleutelstroom.
- Gebruik een sleutelstroom nooit opnieuw met dezelfde sleutel.

3. AES

- Veelgebruikt blokcijfer met blokken van 128 bits.
- Sleutellengten: 128, 192 of 256 bits.

4. DES en 3DES

- DES heeft een effectieve sleutellengte van 56 bits en is niet meer veilig voor moderne toepassingen.
- 3DES past DES herhaaldelijk toe; het is traag en uitgefaseerd voor nieuwe toepassingen.

5. Werkmodi: CBC en GCM

- Een werkmodus bepaalt hoe een blokcijfer langere gegevens verwerkt.
- CBC vereist een onvoorspelbare initialisatievector (IV); CBC-versleuteling alleen beschermt de integriteit niet.
- GCM combineert versleuteling met authenticatie en levert een authenticatietag.
- Herhaal een nonce nooit met dezelfde GCM-sleutel: dat kan de beveiliging ernstig ondermijnen.

6. Sleutelbeheer

- Genereer sleutels veilig; beperk toegang en bewaar sleutels vertrouwelijk.
- Distribueer sleutels veilig; roteer ze waar nodig en trek ze in wanneer ze niet meer vertrouwd zijn.
- Houd ook IV's en nonces correct en uniek waar de modus dat vereist.

7. Prestatie-afwegingen

- Weeg snelheid, hardwareondersteuning, geheugen en gegevensgrootte af.
- Houd rekening met de behoefte aan authenticatie, niet alleen met versleuteling.
- AES met een geauthenticeerde modus zoals GCM is een gangbare keuze als die correct wordt ondersteund en toegepast.

8. Korte samenvatting

- Bescherm de geheime sleutel en gebruik de gekozen modus correct.
- CBC alleen biedt geen integriteitsbescherming; hergebruik geen GCM-nonce met dezelfde sleutel.
- Kies een passende, geauthenticeerde toepassing en beheer sleutels zorgvuldig.

Grondbeginselen van asymmetrische cryptografie

1. Wat is het?

- Twee verschillende sleutels: openbaar en geheim.
- Welke functie ze vervullen hangt af van het algoritme.
- Openbare sleutel: mag worden gedeeld.
- Geheime sleutel: blijft bij de eigenaar.
- Versleuteling, digitale handtekeningen en sleuteluitwisseling zijn verschillende toepassingen.

2. Sleutelparen

- Een sleutel werkt met de bijbehorende sleutel, maar ze zijn niet onderling verwisselbaar.
- Versleutelen voor iemand; gebruik diens openbare sleutel; ontsleutelen vereist de bijbehorende geheime sleutel.
- Ondertekenen: afzender gebruikt doorgaans geheime sleutel; anderen controleren met openbare sleutel.
- Een geldige handtekening ondersteunt controle van herkomst en integriteit; ze versleutelt het bericht niet.

3. Wiskundige basis

- Rekenkundige relaties zijn eenvoudig vooruit te berekenen, maar bepaalde omgekeerde problemen zijn moeilijk.
- Priemgetallen: groter dan 1, alleen deelbaar door 1 en zichzelf.
- Modulo: rekenen met een rest; 17 modulo 5 = 2.
- Machten: herhaald vermenigvuldigen, vaak gecombineerd met modulo.
- Moeilijke omgekeerde problemen helpen de geheime sleutel beschermen. Intuïtieve uitleg, geen formeel bewijs.

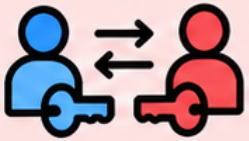
4. RSA

- Gebruikt eigenschappen van grote priemgetallen en modulaire rekenkunde om een sleutelrelatie te construeren.
- Kan worden gebruikt voor versleuteling of digitale handtekeningen, afhankelijk van de toepassing.
- Veiligheid hangt mede af van geschikte sleutels en veilige implementatie.
- De naam RSA alleen zegt niet welke toepassing of instellingen zijn gebruikt.

5. ECC

- Elliptische-krommecryptografie gebruikt rekenkundige bewerkingen op punten van elliptische krommen over eindige velden.
- Toepassingen omvatten sleuteluitwisseling en digitale handtekeningen.
- Afhankelijk van parameters en implementatie kan vergelijkbare beveiligingssterkte met kleinere sleutels dan RSA bieden.
- Parameters en correcte implementatie blijven belangrijk.

6. Diffie-Hellman



- Twee partijen leiden via een openbaar kanaal hetzelfde gedeelde geheim af, zonder het rechtstreeks te verzenden.
- Ze wisselen openbare waarden uit en combineren die met hun eigen geheime waarde.
- Waarden zijn zichtbaar voor anderen; het afgeleide geheim wordt niet rechtstreeks meegestuurd.
- De methode is op zichzelf geen authenticatie.

Let op: Zonder authenticatie kan een aanvaller zich tussen partijen plaatsen en afzonderlijke geheimen met elk afleiden: een **man-in-the-middle-aanval**.

7. Voorbeelden en aandachtspunten

- Vertrouwelijk bericht: versleutel voor ontvanger met diens openbare sleutel; ontvanger ontsleutelt met bijbehorende geheime sleutel.
- Handtekening: afzender ondertekent met geheime sleutel; ontvanger controleert met openbare sleutel.
- Sleuteluitwisseling: partijen leiden met Diffie-Hellman een gedeeld geheim af zonder het rechtstreeks te verzenden.
- Vraag: versleuteling, ondertekening of sleuteluitwisseling? Geen synoniemen.

8. Korte samenvatting

- Openbare, deelbare sleutel + privé gehouden geheime sleutel.
- RSA: priemgetallen en modulo-rekenen.
- ECC: bewerkingen op punten van elliptische krommen.
- Diffie-Hellman: gedeeld geheim afleiden; zonder authenticatie kwetsbaar voor man-in-the-middle-aanval.
- Handtekeningen controleren herkomst en integriteit; ze zijn niet hetzelfde als versleuteling.

Cryptografische hashes & integriteitscontrole

1. Wat is een hash?

- Een hashfunctie zet invoer van willekeurige lengte om in een hashwaarde met vaste lengte.
- Bij een deterministische functie levert dezelfde invoer steeds dezelfde hashwaarde op.
- Zie de hash als een compacte digitale vingerafdruk van de inhoud.

2. SHA-2 en SHA-3

- SHA-2 is een familie met onder meer SHA-256 en SHA-512.
- SHA-3 is een aparte familie, gebaseerd op de Keccak-constructie.
- SHA-2 en SHA-3 zijn verschillende algoritmefamilies.

3. Belangrijke eigenschappen

- **Pre-image-weerstand:** het hoort praktisch onhaalbaar te zijn een invoer te vinden die een gegeven hash oplevert.
- **Botsingsbestendigheid:** het hoort praktisch onhaalbaar te zijn twee verschillende invoeren met dezelfde hash te vinden.
- Een kleine wijziging in de invoer leidt doorgaans tot een andere hashwaarde.

4. Hash is geen versleuteling

- Een hash is geen versleutelde tekst.
- Er is geen sleutel waarmee je de oorspronkelijke invoer uit de hash terughaalt.
- Gebruik versleuteling wanneer gegevens later weer ontsleuteld moeten kunnen worden.

5. Salting en wachtwoorden

- Een salt is een willekeurige, unieke waarde die met een wachtwoord wordt verwerkt vóór het hashen.
- Gebruik voor wachtwoordopslag een daarvoor ontworpen, traag schema, zoals Argon2id, scrypt, bcrypt of PBKDF2.
- Bewaar de salt en de gebruikte parameters bij de wachtwoordhash.
- Een snelle algemene hash zoals SHA-256 is op zichzelf geen geschikte wachtwoordopslagmethode.

6. Bestandsintegriteit

- Bereken lokaal de hash van een bestand en vergelijk die met een waarde uit een vertrouwde, onafhankelijke bron.
- Een overeenkomst ondersteunt dat de inhoud overeenkomt met het bestand waarvoor de hash is gepubliceerd.
- Een overeenkomst bewijst op zichzelf niet dat de bron of het bestand veilig of authentiek is.

7. Praktijkvoorbeeld

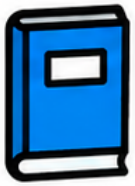
Download
Download het bestand van de officiële bron.

Bereken
Bereken lokaal de SHA-256-hash van het bestand.

Vergelijk
Vergelijk de uitkomst met de SHA-256-waarde die de officiële bron publiceert.

Bij verschil
Controleer de herkomst van de hashwaarde en download het bestand opnieuw. De vergelijking is een integriteitscontrole, geen zelfstandig bewijs van veiligheid.

8. Korte samenvatting



- Hashes zetten invoer om in een waarde met vaste lengte; SHA-2 en SHA-3 zijn aparte families.
- Pre-image-weerstand en botsingsbestendigheid beschrijven verschillende gewenste eigenschappen.
- Hashen is geen versleuteling; bescherm wachtwoorden met een geschikt traag schema en unieke salt.
- Een hashvergelijking kan integriteit ondersteunen, maar bewijst niet op zichzelf veiligheid of authenticiteit.

Digitale handtekeningen & PKI-infrastructuur

1. Wat is het

- PKI (Public Key Infrastructure) is een stelsel van sleutels, certificaten, afspraken en vertrouwensrelaties.
- Een digitale handtekening koppelt gegevens aan een privésleutel en kan aantonen dat gegevens na ondertekening niet zijn gewijzigd.
- Certificaten helpen clients vaststellen bij welke identiteit of welk onderwerp een publieke sleutel hoort.

2. Sleutels en handtekeningen

- Ondertekenaar maakt handtekening met privésleutel.
- Bijbehorende publieke sleutel verifieert de handtekening.
- Privésleutel blijft vertrouwelijk en onder controle van de eigenaar.
- Geldige handtekening ondersteunt integriteitscontrole en toont dat de handtekening bij de gebruikte sleutel hoort.

3. X.509-certificaten

- X.509-certificaat bindt publieke sleutel aan identiteit of onderwerp.
- Bevat onder meer onderwerp, uitgever en geldigheidsperiode.
- Uitgevershandtekening helpt controleren dat certificaat door die uitgever is ondertekend.
- Geen geheime sleutel: publieke sleutel en certificaatgegevens mogen worden gedeeld.

4. CA en uitgifte

- Certificaatautoriteit (CA) geeft certificaten uit en ondertekent ze.
- CA verbindt publieke sleutel met opgegeven identiteit of onderwerp volgens haar uitgifteproces.
- Client vertrouwt certificaat niet alleen omdat het bestaat: handtekening en keten moeten worden gevalideerd.
- Vertrouwen hangt mede af van lokaal ingestelde vertrouwensankers.

5. Vertrouwensketen

- Certificaat kan via tussenliggende certificaten leiden naar vertrouwde anker-CA.
- Client controleert handtekeningen en geldigheidsperioden in de keten.
- Keten moet uitkomen bij lokaal vertrouwde anker-CA.
- Certificaat moet passen bij doel, zoals domeinnaam van HTTPS-server.

HTTPS: browser controleert of servercertificaat geldig is, bij domeinnaam past en via vertrouwde keten kan worden gevalideerd.

6. Intrekking: CRL en OCSP-stapling

- CA-ondertekende certificaatintrekkingslijst (CRL) vermeldt ingetrokken certificaten.
- OCSP biedt een manier om intrekkingsstatus op te vragen.
- Bij OCSP-stapling stuurt TLS-server recent OCSP-antwoord mee dat client kan controleren.
- Beschikbare intrekkingscontrole hangt af van gebruikte mechanismen.

7. Niet-weerlegbaarheid

- Niet-weerlegbaarheid beoogt bewijs aan partij en handeling te koppelen.
- Digitale handtekening alleen garandeert dit niet in elke context.
- Sleutelbeheer, betrouwbare identiteitstoewijzing, proces en toepasselijke regels spelen ook mee.
- Gedeelde of gecompromitteerde privésleutel kan twijfel oproepen over wie handeling uitvoerde.

Onthoud: handtekening bewijst niet vanzelf wie de sleutel gebruikte of dat een handeling juridisch niet kan worden betwist.

8. Korte samenvatting

- Privésleutel: ondertekenen; bijbehorende publieke sleutel: verifiëren.
- X.509-certificaat koppelt publieke sleutel aan onderwerp of identiteit.
- CA en vertrouwensketen helpen client bepalen of koppeling wordt vertrouwd.
- CRL en OCSP: mechanismen om intrekkingsstatus te controleren.
- HTTPS: controleer geldigheid, domeinnaam en keten; handtekening alleen garandeert geen niet-weerlegbaarheid.

TLS/SSL & veilige webcommunicatie

Een overzicht van hoe TLS een webverbinding beveiligt en welke onderdelen daarbij horen.

1 Wat is het

- TLS beveiligt de verbinding tussen client en server.
- SSL is de verouderde voorganger van TLS.

2 Waarom TLS

- Versleutelt gegevens tijdens transport.
- Helpt afluisteren en wijziging onderweg tegen te gaan.
- HTTPS gebruikt HTTP over TLS.

3 Handshake

- Client en server spreken verbindingsparameters af.
- Ze voeren sleuteluitwisseling en serverauthenticatie uit.
- De precieze stappen hangen af van de TLS-versie.

4 Cryptografische suites

- Beschrijven cryptografische algoritmen of functies voor TLS.
- Moderne suites omvatten geauthenticeerde versleuteling.
- De betekenis en samenstelling verschillen per TLS-versie.

5 Certificaatvalidatie

- Controleer keten naar een vertrouwde uitgever, geldigheid en domeinnaam.
- Controleer ook toepasselijke certificaatbeperkingen.
- Een geldig certificaat bewijst niet dat de website betrouwbaar of onschadelijk is.

6 HSTS

- HTTP-responsbeleid dat browsers instrueert volgende verbindingen via HTTPS te maken.
- Beschermt niet tegen alle aanvallen en herstelt geen foutieve HTTPS-configuratie.

7 Configuratiefouten

- Verouderde TLS-versies of zwakke cryptografie toegestaan.
- Certificaat-, domeinnaam- of ketenfouten.
- Gemengde HTTP/HTTPS-inhoud of onjuiste HSTS-instellingen.

Herkenningslijst — Dit zijn voorbeelden van veelvoorkomende aandachtspunten, geen volledige auditstandaard.

8 Korte samenvatting

- TLS beveiligt webcommunicatie; SSL is verouderd.
- Handshake, cryptografische suites en certificaatvalidatie werken samen.
- HSTS helpt browsers HTTPS te blijven gebruiken, maar voorkomt niet elk risico.

Ransomware: levenscyclus en mitigatie

1. Wat is ransomware?

- Kwaadaardige software die toegang tot systemen of bestanden kan blokkeren, vaak door bestanden te versleutelen.
- De aanvaller kan losgeld eisen om toegang te herstellen of het openbaar maken van gestolen gegevens te voorkomen.

2. Initiële toegang

- Phishingbericht of schadelijke bijlage.
- Gestolen inloggegevens.
- Misbruik van een kwetsbare of onbeveiligde dienst.
- Toegang via een getroffen leverancier of andere derde partij.

3. Voet aan de grond en privileges

- Een aanvaller probeert toegang te behouden en kan proberen meer rechten te verkrijgen.
- Beperk accountrechten tot wat iemand nodig heeft.
- Gebruik meervoudige verificatie waar beschikbaar en installeer beveiligingsupdates tijdig.

4. Laterale verplaatsing

- De aanvaller beweegt vanuit een getroffen account of systeem naar andere systemen in dezelfde omgeving.
- Netwerksegmentatie en beperkte accountrechten kunnen de verspreidingsmogelijkheden verkleinen.

5. Gegevens verzamelen en exfiltreren

- Exfiltratie is het ongeoorloofd overbrengen van gegevens uit een omgeving.
- Sommige incidenten combineren gegevensdiefstal met versleuteling; andere niet.

6. Versleuteling en losgeldeis

- Bestanden of systemen kunnen worden versleuteld of op een andere manier ontoegankelijk gemaakt.
- De losgeldeis kan betaling vragen voor herstel of dreigen met openbaarmaking van gegevens.
- Betaling garandeert niet dat bestanden worden hersteld of gestolen gegevens worden verwijderd.

7. Mitigatie en 3-2-1-back-ups

- Werk systemen tijdig bij; gebruik meervoudige verificatie waar beschikbaar.
- Filter e-mail, train gebruikers om verdachte berichten te herkennen en beperk accountrechten.
- Segmenteer netwerken om verspreiding te bemoeilijken.
- 3-2-1: bewaar 3 kopieën van belangrijke gegevens, op minstens 2 soorten opslag, waarvan minstens 1 offline of anderszins geïsoleerd.
- Bescherm back-ups tegen ongeoorloofde wijzigingen en test herstel regelmatig.

8. Herstel en korte samenvatting

- Een herstelplan legt verantwoordelijkheden, prioriteiten, communicate en herstelstappen vast.
- Oefeningen en hersteltests helpen tekortkomingen opsporen.
- Herstel met bekende schone systemen en back-ups; onderzoek de getroffen omgeving voordat systemen opnieuw worden aangesloten.
- Levenscyclus in één oogopslag: toegang → voet aan de grond → verplaatsing → gegevens verzamelen of exfiltreren → versleuteling of verstoring → losgeldeis. Volgorde en onderdelen kunnen verschillen.

Onthoud

Dit is algemene educatieve informatie. Volg bij een incident het incidentresponsproces van je organisatie.

Mindmap 020: DoS- en DDoS-aanvallen en mitigatie

1. Wat is het

- DoS verstoring vanuit één of meer bronnen.
- DDoS vanuit veel bronnen tegelijk, vaak botnet.
- Website/netwerkdienst doelwit.

2. Waarom relevant

- Onbereikbaarheid onderbreekt werk en dienstverlening.
- Gebruikers tijdelijk geen toegang.
- Voorbereiding, monitoring, herstel.

3. Aanvalstypen

- Volumetrisch belast netwerkcapaciteit.
- SYN-flood belast verbindingsafhandeling.
- UDP-reflectie stuurt verkeer via publiek bereikbare diensten.
- NTP-amplificatie gebruikt grotere NTP-antwoorden.
- HTTP-flood belast applicatie.
- Slowloris houdt verbindingen lang open met onvolledige verzoeken.

4. Hoe het werkt

- Effect afhankelijk van laag.
- Netwerkbandbreedte/ verbindingscapaciteit.
- Applicatieverwerking door herhaalde verzoeken.
- Reflectie stuurt verkeer terug naar doelwit.

5. Signalen

- Verkeerspieken/ongebruikelijke patronen.
- Trage of onbereikbare webdienst.
- Veel nieuwe of lang open verbindingen.
- Beoordeel met monitoring en normaal patroon.

6. Mitigatie

- Filtering verdacht verkeer en passende snelheidslimieten.
- Monitor verkeer/beschikbaarheid.
- Plan capaciteit en beschikbaarheid.
- Stem af met provider.

Scrubbing center: leidt verkeer om, filtert schadelijk verkeer en laat legitiem verkeer door.

Stem mitigatie af met beheerder of provider.

7. Praktijkvoorbeeld

- Webwinkel wordt traag in drukke periode.
- Team ziet verkeersafwijking, controleert monitoring/ bereikbaarheid en schakelt provider in.
- Provider filtert verkeer, zo nodig via scrubbing center.
- Controleer legitieme bezoekers weer toegang hebben.

8. Korte samenvatting

- DoS/DDoS verstoren beschikbaarheid.
- Belasten netwerk of applicatie.
- Detectie, filtering, voorbereiding en samenwerking met providers beperken impact.
- Herken aanvalstype en monitor afwijkingen.
- Scrubbing filtert schadelijk verkeer vóór beschermde infrastructuur.

Mindmap 021: Man-in-the-Middle (MitM)-aanvallen en preventie

1. Wat is MitM?

- Een aanvaller onderschept de communicatie tussen twee partijen. Die partijen kunnen denken dat ze rechtstreeks met elkaar communiceren.
- Verkeer kan worden bekeken of gewijzigd.
- Voorbeeld: een nepnetwerk onderschept verbindingen op een openbare wifi-hotspot.

2. Hoe werkt het?

- De aanvaller probeert zich tussen de gebruiker en de bedoelde dienst te plaatsen.
- De verbinding wordt omgeleid of onderschept.
- Daarna kan de aanvaller gegevens observeren of aanpassen.
- Een beveiligde verbinding helpt onderschepping en wijziging tegen te gaan.

3. ARP-spoofing

- ARP koppelt in een lokaal IPv4-netwerk een IP-adres aan een MAC-adres. Bij ARP-spoofing wordt die koppeling vervalst.
- Verkeer binnen hetzelfde lokale subnet kan naar een verkeerd MAC-adres worden gestuurd.
- Voorbeeld: een apparaat leert onverwacht een ander MAC-adres voor de router.
- Een afwijking is een aanwijzing, geen bewijs van een aanval.

4. DNS-spoofing

- DNS vertaalt domeinnamen naar IP-adressen. Bij DNS-spoofing krijgt een apparaat een vervalst antwoord.
- Een domeinnaam kan naar een onbedoeld adres verwijzen.
- Voorbeeld: een bekende domeinnaam verwijst onverwacht naar een onbekende server.
- DNSSEC helpt de authenticiteit en integriteit van DNS-gegevens te controleren; het versleutelt DNS-verkeer niet.

5. SSL-stripping

- SSL-stripping is een downgrade-aanval: een verbinding die HTTPS zou gebruiken, wordt teruggebracht naar HTTP.
- HTTP versleutelt het verkeer niet.
- HSTS helpt een browser HTTPS voor een domein af te dwingen.
- Negeer certificaatwaarschuwingen niet.

6. Detectie

- Let op onverwachte veranderingen en waarschuwingen. Geen enkel signaal bewijst op zichzelf dat er een MitM-aanval plaatsvindt.
- Onverwachte waarschuwing over een certificaat.
- Afwijkend DNS-antwoord of onverwachte wijziging in de ARP-tabel.
- Onbekende apparaten of gateways op het lokale netwerk.
- Meld verdachte signalen aan de netwerkbeheerder.
- Let op: Controleer afwijkingen via een betrouwbare beheerder of netwerkbron. Klik niet zomaar door certificaatwaarschuwingen heen.

7. Netwerklaagbeveiliging

- Combineer veilige verbindingen met passende netwerkcontroles.
- Gebruik HTTPS en controleer certificaatwaarschuwingen.
- Gebruik HSTS en DNSSEC waar passend.
- Beveilig wifi met sterke instellingen en beperk toegang tot het netwerk.
- Segmenteer netwerken zodat apparaten niet onnodig met elkaar communiceren.
- Op ondersteunde switches kunnen DHCP snooping en Dynamic ARP Inspection ongewenste ARP-wijzigingen helpen beperken.

8. Korte samenvatting

- MitM plaatst een aanvaller tussen communicerende partijen. ARP- en DNS-spoofing kunnen verkeer omleiden; SSL-stripping verlaagt de verbinding naar HTTP.
- Let op signalen, maar behandel ze als aanwijzingen.
- Gebruik HTTPS met geldige certificaatcontrole en passende netwerkbeveiliging.

Phishing & spear-phishing

1. Wat is phishing?

- Misleidende berichten of contacten proberen je gevoelige gegevens te laten delen, geld over te maken of iets anders onveiligs te doen.
- Phishing is een vorm van social engineering: iemand probeert je via manipulatie tot actie te bewegen.

2. Spear-phishing

- Gerichte phishing, speciaal afgestemd op één persoon of organisatie.
- Een bericht kan je naam, functie, collega's of een lopend project noemen om geloofwaardig te lijken.
- Een persoonlijk detail bewijst niet dat de afzender betrouwbaar is.

3. Kanalen en tactieken

- E-mail, sms, chat- en berichtenapps of telefoon.
- Phishing via telefoongesprekken heet vaak vishing; via sms heet vaak smishing.
- De afzender kan zich voordoen als een collega, organisatie of dienst.

4. Herkenningssignalen

- Onverwacht verzoek om inloggegevens, betaling of persoonlijke informatie.
- Druk of haast, een ongewoon adres of nummer, of een link die niet bij de genoemde organisatie past.
- Onverwachte bijlage of verzoek om normale procedures over te slaan.
- Eén signaal bewijst niet dat een bericht frauduleus is; verzorgde taal bewijst evenmin dat het echt is.

5. Links en bijlagen controleren

- Vergelijk het volledige afzenderadres met het verwachte domein.
- Bekijk de bestemming van een link zonder deze te openen; let op afwijkende domeinnamen.
- Vraag je af of het verzoek past bij de situatie en de normale werkwijze.
- Controleer belangrijke verzoeken via een apart, bekend en vertrouwd kanaal.

6. Veilig reageren

- Antwoord niet op een verdacht bericht en open onverwachte bijlagen niet.
- Voer geen inloggegevens in via een link uit het bericht.
- Neem bij twijfel zelf contact op via bekende contactgegevens of de gebruikelijke procedure.

7. Melden en opvolgen

- Meld het bericht via de aangewezen meldmethode of neem via een bekend kanaal contact op met IT of informatiebeveiliging.
- Bewaar het bericht en relevante gegevens als dat veilig kan; volg de interne procedure.
- Heb je geklikt, een bijlage geopend, gegevens gedeeld of geld overgemaakt? Meld dit meteen via het incidentproces.
- Wijzig blootgestelde wachtwoorden via de officiële dienst. Neem bij mogelijk financieel risico contact op met je bank of dienstverlener.

8. Korte samenvatting

- Pauzeer bij onverwachte verzoeken of druk.
- Controleer afzender, link en context; verifieer belangrijke verzoeken apart.
- Reageer niet via verdachte berichten en meld vermoedelijke phishing.
- Meld een mogelijke blootstelling direct; snel melden helpt de organisaite opvolgen.

023. Geavanceerde phishing: whaling, vishing en smishing

1. Wat is het?

- Phishing: misleidend contact dat aanzet tot gegevens delen, geld overmaken of een andere actie.
- Kan via e-mail, chat, telefoon of sms komen.
- Whaling, vishing en smishing zijn vormen van phishing.

2. Whaling

- Gerichte phishing die leidinggevenden of andere invloedrijke personen imiteert of benadert.
- Maakt mogelijk misbruik van hun toegang of bevoegdheid.
- Voorbeeld: een ongebruikelijk verzoek om met spoed een betaling uit te voeren.

3. Vishing

- Misleiding via telefoon-gesprekken of gesproken berichten.
- Een beller kan zich voordoen als een vertrouwd persoon of een organisatie.
- Deel geen wachtwoord, pincode of eenmalige code op verzoek van een onverwachte beller.

4. Smishing

- Misleiding via sms of andere tekstberichten.
- Een bericht kan aandringen op een betaling, gegevens of het openen van een link.
- Gebruik links en contactgegevens uit een verdacht bericht niet om het verzoek te controleren.

5. Herkenningssignalen

- Onverwachte haast, geheimhouding of druk om normale procedures over te slaan.
- Verzoek om wachtwoorden, pincodes, eenmalige codes of geld.
- Ongebruikelijke afzendergegevens, links of bijlagen.

Let op Een signaal is reden om te controleren, maar bewijst op zichzelf niet dat een bericht frauduleus is.

6. Verificatieprotocol

Stop
Voer de gevraagde actie niet meteen uit.

Gebruik geen berichtlink
Klik niet op links en gebruik geen contactgegevens uit het verdachte bericht.

Controleer onafhankelijk
Neem via een eerder vertrouwd kanaal contact op, bijvoorbeeld met een bekend nummer of het interne contactregister.

Volg de goedkeuring
Controleer gevoelige of financiële opdrachten volgens de normale goedkeurings- en controleprocedures.

7. Bij twijfel of een incident

- Antwoord niet en deel geen gegevens of codes zolang het verzoek niet onafhankelijk is gecontroleerd.
- Meld een verdacht verzoek via het beveiligings- of incidentmeld-kanaal van de organisatie.
- Is er al geld of informatie verstuurd? Meld dit zo snel mogelijk via dat kanaal.

8. Korte samenvatting

- Verifieer een onverwacht verzoek via een onafhankelijk, vertrouwd kanaal.
- Deel nooit wachtwoorden, pincodes of eenmalige codes op verzoek van onverwacht contact.

SQL-injectie (SQLi)

1. Wat is SQLi?

- SQL-injectie (SQLi) ontstaat wanneer gebruikersinvoer als SQL-code wordt verwerkt in plaats van uitsluitend als data.
- Daardoor kan de betekenis van een databasequery veranderen.
- Invoer is niet automatisch SQLi: het risico hangt af van hoe de applicatie de query opbouwt.

2. Hoe ontstaat het?

- De applicatie bouwt een query op met gegevens die een gebruiker heeft ingevoerd.
- Als invoer rechtstreeks in de SQL-tekst terechtkomt, kan die de bedoelde querystructuur beïnvloeden.
- Veilige aanpak: houd querystructuur en gebruikerswaarden gescheiden.

3. Kwetsbare SQL-constructies

- SQL-tekst samenstellen door gebruikersinvoer aan een query vast te plakken.
- Invoer direct in een SQL-expressie interpoleren.
- Dynamische queryonderdelen gebruiken zonder veilige controle.
- Voorbeeld van het risico: een zoekquery die tekst uit een invoerveld rechtstreeks aan de SQL-query toevoegt.

4. Risicovolle invoerpatronen

- Aanhalingstekens, SQL-syntaxis en booleaanse expressies kunnen een kwetsbare query beïnvloeden.
- Dezelfde invoer hoeft geen probleem te zijn als de applicatie die veilig als data behandelt.
- Beoordeel daarom de queryopbouw; leid kwetsbaarheid niet alleen af uit de invoer.

5. Gevolgen

- Een gewijzigde query kan anders werken dan de ontwikkelaar bedoelde.
- Mogelijke gevolgen hangen af van de query, de applicatie en de toegangsrechten van de databasegebruiker.
- Beperk risico door veilige queryopbouw en passende databaseprivileges.

6. Geparametriseerde queries

- Schrijf de SQL-structuur met placeholders.
- Geef gebruikerswaarden apart mee via de parameterinterface van de databasebibliotheek.
- De database behandelt die waarden als data, niet als SQL-code.
- Parameteriseer waarden in plaats van invoer in SQL-tekst samen te voegen.

Kernmaatregel: Geparametriseerde queries zijn de primaire verdediging tegen SQL-injectie.

7. Extra verdedigingslagen

- Gebruik voor de databasegebruiker alleen de minimale benodigde privileges.
- Toon gebruikers generieke foutmeldingen; leg details vast in beheerlogs.
- Gebruik voor dynamische identifiers of queryonderdelen die niet als parameter kunnen worden doorgegeven vaste toegestane keuzes en een allowlist.
- Deze lagen vullen geparametriseerde queries aan, maar vervangen ze niet.

8. Korte samenvatting

- SQLi ontstaat wanneer invoer als code kan worden verwerkt en de querybetekenis kan veranderen.
- Plak gebruikersinvoer niet in SQL-tekst en interpoleer die niet rechtstreeks in SQL-expressies.
- Parameteriseer waarden; beperk dynamische keuzes met een allowlist.
- Verklein de mogelijke impact met minimale privileges, generieke foutmeldingen en logging.

Cross-Site Request Forgery (CSRF) & toestandsvervalsing

1. Wat is het

- Een vervalst verzoek vanuit een vertrouwde, ingelogde browsersessie.
- Misbruikt dat de browser vaak automatisch sessiecookies meestuurt.
- Geen wachtwoorddiefstal en geen uitlezen van de website-respons.

2. Hoe werkt het

- Je bent ingelogd bij een website; je browser bewaart een sessiecookie.
- Een andere site laat je browser een verzoek naar die website versturen.
- De browser kan de cookie meesturen; de server ontvangt het verzoek.
- Zonder voldoende controle kan de server het als jouw opdracht verwerken.

3. Waarom kwetsbaar

- De browser koppelt cookies automatisch aan verzoeken naar hun website.
- De server vertrouwt daardoor mogelijk op de sessie zonder de bedoeling te controleren.
- Vooral verzoeken die gegevens of instellingen wijzigen vragen om passende controles.

4. Anti-CSRF-tokens

- Een onvoorspelbare waarde, gekoppeld aan de sessie of het formulier.
- De server controleert de token bij relevante verzoeken.
- Een andere site kan die waarde doorgaans niet uitlezen door same-origin-beveiliging.
- Ontbreekt een geldige token? Wijs het verzoek af.

5. SameSite-cookies

- Bepaalt wanneer een browser cookies meestuurt bij verzoeken vanaf een andere site.
- Strict beperkt dit het meest; Lax laat cookies toe bij bepaalde navigaties op topniveau.
- None staat cross-site gebruik toe en vereist Secure.
- Bescherming hangt af van cookie-instellingen en verzoekcontext; SameSite is een extra laag, geen volledige vervanging voor tokencontrole.

6. Extra verdedigingslagen

- Controleer op de server of wijzigingsverzoeken bij de ingelogde sessie horen.
- Controleer waar passend de Origin- of Referer-header.
- Gebruik passende methoden voor wijzigingen; verander geen gegevens via veilige leesverzoeken.
- Combineer controles passend bij de toepassing.

7. Herkenbaar voorbeeld

- Je bent ingelogd en wilt je e-mailadres wijzigen.
- De website vraagt een wijzigingsverzoek met een geldige CSRF-token.
- De server controleert de token vóór het opslaan.
- Zonder geldige token wordt de wijziging geweigerd.

8. Korte samenvatting

- CSRF misbruikt een bestaande vertrouwde browsersessie om een verzoek te laten versturen.
- De server moet wijzigingsverzoeken controleren, bijvoorbeeld met anti-CSRF-tokens.
- SameSite beperkt het meesturen van cookies vanaf andere sites, maar is niet de enige bescherming.

Wachtwoordaanvallen

1. Wat is het?

Wachtwoordaanvallen proberen toegang te krijgen door wachtwoorden te raden of eerder buitgemaakte inloggegevens opnieuw te gebruiken.

- **Brute force**: systematisch mogelijke wachtwoordcombinaties proberen.
- **Woordenboekaanval**: veelvoorkomende wachtwoorden of kandidaten uit woordenlijsten proberen.
- **Credential stuffing**: eerder buitgemaakte combinaties van gebruikersnaam en wachtwoord opnieuw proberen.

2. Brute force

Bij brute force worden mogelijke wachtwoorden systematisch geprobeerd. Zwakke, korte of voorspelbare wachtwoorden zijn kwetsbaarder voor raden.

- Elke poging test een mogelijke combinatie.
- Een lang, uniek wachtwoord maakt raden moeilijker.
- Herhaalde mislukte aanmeldingen kunnen een waarschuwingssignaal zijn.

3. Woordenboekaanval

Bij een woordenboekaanval worden kandidaten uit lijsten met woorden of veelvoorkomende wachtwoorden gebruikt, in plaats van alle mogelijke combinaties.

- Voorspelbare woorden en wachtwoordpatronen zijn risicovol.
- Een uniek wachtwoord dat niet op een bekend woord of patroon leunt, verkleint dit risico.
- Het is een andere methode dan systematisch alle combinaties proberen.

4. Credential stuffing

Credential stuffing gebruikt combinaties van gebruikersnamen en wachtwoorden die eerder bij een datalek zijn buitgemaakt, om toegang tot andere diensten te proberen.

- Wachtwoordhergebruik vergroot het risico.
- Voorbeeld: een eerder gelekt wachtwoord wordt opnieuw geprobeerd bij een andere dienst.
- Gebruik voor iedere dienst een eigen wachtwoord.

5. Hoe verschillen de methoden?

De drie methoden verschillen vooral in de herkomst van de wachtwoordpogingen.

- **Brute force**: mogelijke combinaties systematisch proberen.
- **Woordenboekaanval**: kandidaten uit woordenlijsten of veelvoorkomende wachtwoorden proberen.
- **Credential stuffing**: eerder gelekte inlogcombinaties opnieuw gebruiken.

6. Signalen en risico's

Let op onverwachte aanmeldingen en herhaalde mislukte pogingen. Eén signaal bewijst op zichzelf niet dat een aanval plaatsvindt.

- **Meldingen** over aanmelden vanaf een onbekend apparaat of een ongebruikelijke locatie.
- Veel mislukte aanmeldpogingen in korte tijd.
- Een melding dat je wachtwoord voorkomt in informatie over een datalek.
- Wachtwoordhergebruik kan ervoor zorgen dat een lek bij de ene dienst andere accounts raakt.

7. Tegenmaatregelen

Combineer maatregelen om accounts beter te beschermen. Instellingen voor pogingen en vergrendeling moeten ook rekening houden met misbruik en herstel.

- Gebruik voor iedere dienst een uniek, sterk wachtwoord; een wachtwoordmanager kan helpen.
- Schakel multifactorauthenticatie in waar dat beschikbaar is.
- Beperk aanmeldpogingen en gebruik waar passend vertraging of snelheidsbeperking.
- Detecteer verdachte aanmeldingen en informeer gebruikers over relevante gebeurtenissen.
- Overweeg accountvergrendeling na herhaalde mislukte pogingen, met een veilige manier om toegang te herstellen.

Bij een verdachte aanmelding: Wijzig het wachtwoord van het getroffen account en van andere accounts waar hetzellde wachtwoord is gebruikt. Schakel multifactorauthenticatie in en controleer de aanmeldactiviteit.

8. Korte samenvatting

Herken de methode en kies passende verdedigingsmaatregelen.

- **Brute force** probeert combinaties; een woordenboekaanval probeert bekende kandidaten.
- **Credential stuffing** hergebruikt gelekte inlogcombinaties; wachtwoordhergebruik vergroot het risico.
- Unieke wachtwoorden, multifactorauthenticatie en passende aanmeldbeperkingen helpen accounts te beschermen.

Zero-daykwetsbaarheden & patchmanagement

1. Wat is het

- **Zero-daykwetsbaarheid**: een softwarefout die nog onbekend is bij de verantwoordelijke partij, of waarvoor nog geen tijd was om een oplossing beschikbaar te maken.
- Zero-day-exploit: code of een techniek waarmee iemand de fout probeert te misbruiken.
- 'Zero-day' zegt niet automatisch iets over de ernst van de fout.

2. Levenscyclus

- Fout bestaat → ontdekking of melding → beoordeling en afstemming.
- Ontwikkeling en verspreiding van een oplossing of tijdelijke maatregel.
- Een aanvaller kan de fout al misbruiken voordat een patch beschikbaar is.
- Patch beschikbaar → testen → gefaseerd uitrollen → controleren op herstel.

3. Ontdekking en melding

- Een fout kan worden ontdekt door onderzoekers, leveranciers, beheerders of aanvallers.
- Leg product, versie, getroffen functie en waargenomen omstandigheden vast.
- Meld de fout via een passend kanaal bij de leverancier of verantwoordelijke partij.
- Beoordeling en afstemming helpen bepalen welke systemen geraakt zijn en wat nodig is.

4. Analyse en prioriteit

- Breng getroffen producten en versies en blootgestelde systemen in kaart.
- Beoordeel bereikbaarheid, mogelijke impact en beschikbare beveiligingsmaatregelen.
- Prioriteer op ernst, blootstelling, bedrijfsimpact en aanwijzingen voor actief misbruik.
- Herzie de prioriteit wanneer nieuwe informatie beschikbaar komt.

5. Exploitatie en signalen

- Een aanvaller kan een fout misbruiken voordat een patch beschikbaar is.
- Mogelijke signalen zijn onverwacht gedrag, afwijkende verzoeken of onbekende wijzigingen.
- Onderzoek relevante logboeken en meldingen; een signaal op zichzelf bewijst geen misbruik.
- Deel bevindingen via de afgesproken beveiligings- en incidentprocessen.

6. Patch en tijdelijke maatregelen

- Volg leveranciersmeldingen en beoordeel of de patch past bij de aanwezige versie.
- Test de update waar haalbaar en plan installatie en herstelmogelijkheid.
- Zonder patch kunnen tijdelijke maatregelen passend zijn: een functie uitschakelen, blootstelling of toegang beperken, of detectie en monitoring aanscherpen.
- Tijdelijke maatregelen kunnen risico beperken, maar garanderen niet dat een systeem veilig is.

7. Gecontroleerde uitrol

- Houd rekening met kritieke systemen, afhankelijkheden, onderhoudsvensters, back-ups en mogelijke uitval.
- Begin waar passend met een beperkte testgroep en schaal na controle gefaseerd op.
- Controleer na installatie versies, werking en relevante logboeken.
- Leg uitzonderingen en resterende risico's vast.
- Voorbeeld — publiek bereikbare webserver: inventariseer de versies en blootstelling, pas zo nodig tijdelijke maatregelen toe, test de leverancierspatch, rol die gefaseerd uit en controleer daarna versies, logboeken en functionaliteit.

8. Korte samenvatting

- Inventariseer wat geraakt kan zijn en beoordeel blootstelling en impact.
- Beperk risico tijdelijk wanneer nodig; test en plan de patch zodra die beschikbaar is.
- Rol de patch gecontroleerd uit en controleer daarna of herstel en werking zijn bevestigd.

APT's & Cyber Kill Chain

1 Wat is het

Een APT is een doelgerichte, doorgaans langdurige cybercampagne of tegenstander die over middelen beschikt om toegang te verkrijgen en te behouden. De term duidt niet op één specifieke malwarefamilie.

- APT: geavanceerde, aanhoudende dreiging.
- De Cyber Kill Chain helpt een mogelijke campagne te ordenen en te analyseren.

2 Doel en kenmerken

Bij een APT-campagne kan een tegenstander zich op een organisatie of doel richten en proberen langere tijd toegang te houden.

- Doelgericht in plaats van willekeurig.
- Kan uit meerdere fasen bestaan.
- Toegang behouden kan deel uitmaken van de campagne.

3 De zeven fasen

De Cyber Kill Chain beschrijft zeven mogelijke fasen van een aanvalscampagne.

1. **Verkenning** — informatie over een mogelijk doel verzamelen.
2. **Bewapening** — aanvalsmiddelen voorbereiden.
3. **Aflevering** — middelen bij het doel proberen te brengen.
4. **Exploitatie** — een kwetsbaarheid of andere gelegenheid benutten.
5. **Installatie** — toegang of aanwezigheid op een systeem vestigen.
6. **Command-and-control** — communicatie met een gecompromitteerd systeem onderhouden.
7. **Acties op doelen** — activiteiten uitvoeren die het doel van de campagne dienen.

4 Waarneembare signalen

Signalen kunnen aanleiding geven tot nader onderzoek, maar bewijzen op zichzelf geen APT-aanwezigheid.

- Een gerichte phishingpoging kan passen bij aflevering.
- Misbruik van een kwetsbare applicatie kan passen bij exploitatie.
- Periodiek verdacht uitgaand verkeer kan aanleiding geven tot onderzoek naar command-and-control.
- Onverwachte systeem- of accountactiviteiten verdienen contextuele beoordeling.

Let op
Beoordeel signalen in samenhang met loggegevens, systemen en normale activiteiten van de organisatie.

5 Verdediging per fase

Koppel preventieve maatregelen en observaties aan de onderdelen van de mogelijke campagne.

- **Verkenning:** beperk onnodige blootstelling van systemen en informatie.
- **Aflevering:** filter verdachte berichten en downloads.
- **Exploitatie:** patch bekende kwetsbaarheden en gebruik endpointbescherming.
- **Installatie en behoud van toegang:** beperk rechten en registreer systeemactiviteiten.
- **Command-and-control:** monitor uitgaand netwerkverkeer.
- **Acties op doelen:** segmenteer netwerken en onderzoek afwijkende activiteiten.

6 Detectie en analyse

Gebruik meerdere bronnen om een mogelijk incident te onderzoeken en bevindingen in context te plaatsen.

- Bekijk endpoint- en systeemlogboeken.
- Vergelijk netwerkverkeer met het verwachte patroon.
- Leg tijdstippen en betrokken systemen naast elkaar.
- Onderzoek signalen als aanwijzingen, niet als zelfstandig bewijs.

7 Beperkingen en aandachtspunten

De Cyber Kill Chain is een analysekader, geen verplicht of volledig stappenplan.

- Een campagne kan fasen overslaan, herhalen of anders uitvoeren.
- Niet elke aanval volgt dezelfde volgorde.
- Een los signaal bewijst geen APT.
- Gebruik het model samen met andere relevante informatie.

8 Korte samenvatting

APT's zijn doelgerichte, doorgaans langdurige dreigingen. De Cyber Kill Chain biedt zeven fasen om mogelijke campagnes te analyseren en defensieve observaties en maatregelen te ordenen.

- Herken de zeven fasen als analysekader.
- Koppel signalen aan context en onderzoek ze zorgvuldig.
- Beperk blootstelling, patch kwetsbaarheden, beperk rechten, segmenteer en monitor.

MITRE ATT&CK: tactieken en technieken

1. Wat is ATT&CK?

Een **kennisbank** die waargenomen gedrag van aanvallers ordent. Zo kunnen beveiligingsteams bevindingen op een gedeelde manier bespreken.

- Beschrijft gedrag dat in de praktijk is waargenomen.
- Is geen volledige lijst van alle mogelijke aanvallen.

2. Tactiek

Een **tactiek** beschrijft het doel van een aanvaller op een bepaald moment: het 'waarom' achter gedrag.

- Voorbeeld: Initiële toegang (Initial Access) is gericht op het verkrijgen van een eerste voet aan de grond.

3. Techniek

Een **techniek** beschrijft in algemene zin hoe gedrag een tactisch doel kan dienen. Een **subtechniek** beschrijft een specifiekere vorm van een techniek.

- Eén techniek kan bij meer dan één tactiek passen.
- Tactiek = doel; techniek = gedrag.

4. Van toegang tot impact

De **tactieken** vormen een bruikbaar denkkader, geen vaste route die elk incident volgt.

- Initiële toegang: een eerste voet aan de grond verkrijgen.
- Uitvoering, persistentie, privilegeverhoging en verdedigingsontwijking: andere mogelijke doelen.
- Referenties voor toegang tot inloggegevens, ontdekking, laterale beweging, verzamelen en commandovoering kunnen helpen gedrag te ordenen.
- Exfiltratie: gegevens naar buiten brengen. Impact: systemen of gegevens manipuleren, verstoren of vernietigen.

Onthoud | Activiteit kan tactieken overslaan, herhalen of in een andere volgorde volgen.

5. Voorbeeld

Een organisatie onderzoekt verdachte activiteit en gebruikt ATT&CK om waargenomen gedrag te ordenen.

- Het team koppelt bevindingen aan relevante technieken.
- Die indeling helpt het team detecties en verdedigingsmaatregelen te beoordelen.

6. Verdedigingswaarde

ATT&CK biedt een gedeelde structuur om waarnemingen te bespreken en de verdediging te herzien.

- Orden gedrag dat tijdens onderzoek is waargenomen.
- Gebruik de indeling om detecties te structureren en verdedigingsmaatregelen te beoordelen.
- Informatie over procedures, maatregelen en detecties ondersteunt de kennisbank; dit zijn geen extra tactieken.

7. Grenzen en zorgvuldig gebruik

Gebruik ATT&CK als hulpmiddel om waargenomen gedrag te beschrijven, niet als bewijs dat een bepaald verloop of doel zeker is.

- De kennisbank omvat niet elke mogelijke aanval.
- Behandel de tactieken niet als verplichte chronologie.
- Maak onderscheid tussen wat is waargenomen en wat je daaruit afleidt.

8. Korte samenvatting

ATT&CK helpt teams waargenomen aanvallersgedrag te ordenen en er samen over te communiceren.

- Tactiek = doel of 'waarom'.
- Techniek = gedrag of 'hoe'; subtechniek = specifickere vorm.
- Gebruik de indeling voor gedeeld begrip en verdedigingsreview, niet als vaste route.

Netwerksegmentatie & VLAN's

1. Wat is het

- Netwerksegmentatie verdeelt een netwerk in afzonderlijke zones.
- Een VLAN (Virtual LAN) is een logische indeling die apparaten in aparte Layer 2-netwerken kan plaatsen, ook op dezelfde fysieke switch.

2. Logische zones

- Deel bijvoorbeeld in op functie: medewerkers, servers en gasten.
- Koppel alleen de zones die voor een taak met elkaar moeten communiceren.

3. Broadcastdomeinen

- Een VLAN vormt doorgaans een eigen Layer 2-broadcastdomein.
- Broadcastverkeer binnen één VLAN wordt niet automatisch naar andere VLAN's doorgestuurd.

4. VLAN-werking

- Een accesspoort is doorgaans aan één VLAN toegewezen; eindapparatuur ontvangt meestal ongelabelde frames.
- Op een trunkverbinding kan een VLAN-tag aangeven bij welk VLAN een Ethernet-frame hoort.

5. Verkeer tussen VLAN's

- Verkeer tussen VLAN's vereist routering op Layer 3, bijvoorbeeld via een router of Layer 3-switch.
- Toegangsregels op die route bepalen welke verbindingen zijn toegestaan.

6. Beveiligingsvoordeel

- Beperkte communicatiepaden kunnen de impact van ongewenste toegang verkleinen.
- Segmentatie met passende toegangsregels kan zijwaartse beweging bemoeilijken: uitbreiding van toegang na een eerste positie in het netwerk.
- Het voorkomt aanvallen niet op zichzelf.

7. Voorbeeld & aandachtspunten

- Voorbeeld: drie zones — medewerkers, servers en gasten.
- Een gastapparaat bereikt serverdiensten niet rechtstreeks, tenzij routerings- en toegangsregels die verbinding toestaan.
- Een VLAN-indeling alleen is geen volledige beveiligingsgrens.

8. Korte samenvatting

- VLAN's scheiden Layer 2-broadcastdomeinen; segmentatie ordent apparaten in zones.
- Layer 3-routering verbindt VLAN's; toegangsregels begrenzen het verkeer.
- Goede segmentatie beperkt risico's, maar biedt geen garantie.

Firewalls: typen, werking en generaties

1 Wat is het?

- Een firewall controleert netwerkverkeer aan de hand van regels. Zo kan verkeer worden toegestaan of geblokkeerd.
- Beschermt een netwerkgrens of apparaat.
- Regels bepalen welk verkeer wel en niet doorgaat.

2 Pakketfiltering

- Een pakketfilter beoordeelt netwerkpakketten op kenmerken uit hun headers.
- Kenmerken: bron- en bestemmingsadres, poort en protocol.
- Regels laten passend verkeer door of blokkeren het.

3 Stateful inspectie

- Stateful inspectie houdt bij welke verbindingen actief zijn en beoordeelt pakketten in de context van zo'n verbinding.
- Verbindingstoestand helpt bepalen of een pakket bij verwacht verkeer hoort.
- Pakketten worden dus niet ieder volledig los beoordeeld.

4 NGFW

- Een next-generation firewall (NGFW) bouwt voort op traditionele firewallfuncties en kan extra beveiligingsmogelijkheden combineren.
- Mogelijke toevoegingen zijn inbraakpreventie, applicatieherkenning en filtering van domeinen of URL's.
- Welke functies beschikbaar zijn, verschilt per product.

5 Diepe pakketinspectie

- Diepe pakketinspectie (DPI) analyseert pakketinhoud verder dan alleen netwerk- en transportheaders.
- Kan aanvullende informatie over verkeer opleveren.
- Versleuteling kan de zichtbaarheid in pakketinhoud beperken.

6 Mogelijkheden

- Afhankelijk van de implementatie kan een NGFW verschillende functies samenbrengen.
- Applicaties herkennen of verkeer op basis van domein of URL filteren.
- Inbraakpreventie en andere beveiligingsfuncties combineren.
- Context gebruiken, zoals wie verbinding maakt en vanaf welk apparaat.

Let op
Een functienaam garandeert niet dat iedere dreiging wordt gedetecteerd of geblokkeerd.

7 Grenzen en voorbeeld

- Firewallregels werken volgens het ingestelde beleid; inspectie heeft grenzen, vooral wanneer inhoud versleuteld is.
- Voorbeeld: laat webverkeer naar een toegestane dienst door.
- Blokkeer ander verkeer volgens het netwerkbeleid.
- Controleer welke verkeerskenmerken en inhoud daadwerkelijk zichtbaar zijn.

8 Korte samenvatting

- Pakketfilters beoordelen pakketkenmerken; stateful inspectie houdt rekening met verbindingen; NGFW's kunnen extra mogelijkheden combineren, waaronder DPI.
- Deze capaciteiten bestaan naast elkaar en vervangen elkaar niet overal strikt.
- Functies en zichtbaarheid verschillen per product en situatie.

Intrusiedetectie en -preventie (IDS/IPS)

1 Wat is het?

Een IDS/IPS bewaakt netwerkverkeer of gebeurtenissen op apparaten om mogelijk kwaadaardig of afwijkend gedrag te herkennen.

- **IDS:** signaleert en meldt; blokkeert niet noodzakelijk.
- **IPS:** kan volgens ingesteld beleid verdachte verbindingen blokkeren of andere preventieve acties uitvoeren.

2 IDS versus IPS

Het verschil zit vooral in de reactie op een detectie.

- IDS → waarnemen en waarschuwen.
- IPS → analyseren en mogelijk ingrijpen.
- Een IPS kan doorgaans ook detecteren; een IDS voorkomt niet vanzelf verkeer.

3 Waar wordt gemeten?

Systemen kunnen verkeer op netwerkniveau bekijken of gebeurtenissen op afzonderlijke apparaten verzamelen.

- **Netwerkgebaseerd:** analyseert netwerkverkeer; een IDS kan passief meekijken via een kopie van het verkeer.
- **Hostgebaseerd:** verzamelt bijvoorbeeld systeem- of toepassingslogboeken op een apparaat.
- Netwerk- en hostgegevens kunnen elkaar aanvullen.

4 Signatuurdetectie

Vergelijkt waargenomen kenmerken met bekende patronen of regels.

- Kan bekende bedreigingen herkennen wanneer er een passende signatuur is.
- Een onbekende of gewijzigde variant kan buiten de bekende patronen vallen.
- Regels en updates beïnvloeden wat wordt herkend.

5 Anomaliedetectie

Vergelijkt gedrag met een vastgestelde basislijn en zoekt naar afwijkingen.

- Kan ook ongebruikelijke patronen signaleren die niet bij een bekende signatuur passen.
- Een afwijking is niet automatisch kwaadwillig: legitieme veranderingen kunnen fout-positieve meldingen veroorzaken.
- Beoordeel meldingen in hun context.

6 Detectie en preventie in de praktijk

De werking hangt af van zichtbaarheid, regels en configuratie.

- Een netwerk-IDS kan verkeer passief observeren; een netwerk-IPS staat doorgaans in het verkeerspad om zo nodig te kunnen ingrijpen.
- Een IDS meldt bijvoorbeeld herhaalde mislukte aanmeldingen.
- Een IPS kan, afhankelijk van het beleid, het bijbehorende verkeer tijdelijk blokkeren.

7 Voorbeelden en aandachtspunten

Een melding is een aanwijzing om te onderzoeken, geen conclusie op zichzelf.

- Een plotselinge sterke stijging van uitgaand verkeer kan als anomalie worden gemeld; controleer of er een legitieme verklaring is.
- Versleuteling beperkt wat een systeem zonder ontsleuteling kan inspecteren, maar sluit alle detectie niet uit.
- Zichtbaarheid van verkeer, kwaliteit van regels en configuratie beïnvloeden de detectie.

8 Korte samenvatting

IDS signaleert; IPS kan preventief ingrijpen. Beide kunnen netwerk- of hostinformatie gebruiken en verschillende analysemethoden combineren.

- Signaturen zoeken naar bekende patronen; anomaliedetectie zoekt naar afwijkingen ten opzichte van een basislijn.
- Beoordeel meldingen met context: afwijkend gedrag kan ook legitiem zijn.

VPN's en veilige tunnels

1 Wat is een VPN?

- VPN betekent virtueel privénetwerk: een logische privéverbinding over een bestaand netwerk, vaak het internet.
- Een VPN kan netwerklocaties met elkaar verbinden of een gebruiker op afstand toegang geven tot een netwerk of dienst.

2 Waarom een tunnel?

- Een VPN-tunnel kapselt netwerkverkeer in voor vervoer tussen twee tunneluiteinden.
- Een tunnel op zichzelf garandeert geen beveiliging. Het gekozen protocol en de configuratie bepalen welke bescherming wordt geboden.

3 Tunnelprotocollen

- Protocollen bepalen hoe verkeer wordt beschermd en hoe de tunneluiteinden elkaar kunnen authenticeren.
- Versleuteling en peer-authenticatie zijn afhankelijk van het protocol en de instellingen; controleer welke functies de gekozen oplossing daadwerkelijk biedt.

4 IPsec

- IPsec is een reeks netwerklaagprotocollen voor beveiligde communicatie met IP.
- ESP kan, afhankelijk van de configuratie, vertrouwelijkheid, integriteitsbescherming en gegevensbronauthenticatie bieden.
- Tunnelmodus kapselt het oorspronkelijke IP-pakket in een nieuw IP-pakket in. Transportmodus beschermt hoofdzakelijk de inhoud van het IP-pakket.

5 SSL/TLS-VPN

- SSL/TLS-VPN is een gangbare naam voor toegang op afstand waarbij TLS de communicatie tussen een client en een VPN-gateway beschermt.
- TLS beveiligt het communicatiekanaal, maar bepaalt op zichzelf geen VPN-topologie of toegangsbeleid.
- Afhankelijk van de oplossing kan toegang beperkt zijn tot specifieke toepassingen of een bredere netwerkverbinding omvatten.

6 Verbindingsvormen

- Site-to-site: VPN-gateways verbinden netwerken op verschillende locaties.
- Toegang op afstand: een apparaat van een gebruiker verbindt met een bedrijfsnetwerk of geselecteerde interne diensten.
- Voorbeeld: een medewerker gebruikt een VPN om toegang te krijgen tot een interne toepassing.

7 Beveiliging en aandachtspunten

- Gebruik betrouwbare authenticatie en zorg voor een juiste configuratie.
- Geef gebruikers alleen toegang tot de netwerken en toepassingen die zij nodig hebben.
- Houd eindapparaten veilig: een VPN maakt een kwetsbaar apparaat niet vanzelf veilig.
- Een VPN voorkomt niet alle aanvallen.

Onthoud De bescherming hangt af van het protocol, de configuratie, de authenticatie, de toegangsrechten en de veiligheid van het eindapparaat.

8 Korte samenvatting

- Een VPN verbindt gebruikers of netwerken via een tunnel over een bestaand netwerk.
- IPsec en SSL/TLS-VPN zijn verschillende benaderingen; hun mogelijkheden hangen af van de oplossing en configuratie.
- Passende instellingen, betrouwbare toegangscontrole en veilige apparaten zijn belangrijk voor VPN-beveiliging.

DMZ en bastionhosts

1. Wat is het

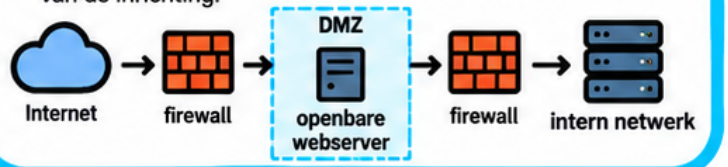
- **DMZ:** afgescheiden netwerksegment voor diensten die vanaf internet bereikbaar zijn.
- **Bastionhost:** extra beveiligde computer als gecontroleerd toegangspunt tot afgeschermde systemen.
- **Verschil:** de DMZ is een netwerkdeel; de bastionhost is een systeem.

2. Waarom gebruiken

- Beperk rechtstreekse verbindingen tussen openbare diensten en het interne netwerk.
- Beheer toegang tot interne systemen via een gecontroleerd toegangspunt.
- Beperk de gevolgen van een incident; segmentatie voorkomt niet elke compromittering.

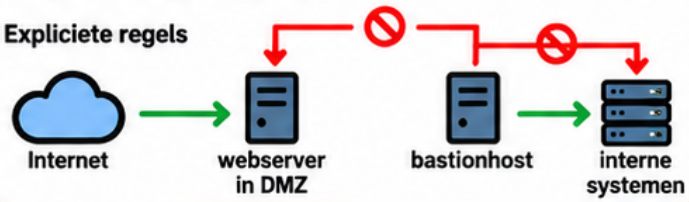
3. Netwerkopbouw

- Internet ↔ firewall ↔ DMZ ↔ firewall ↔ intern netwerk.
- Plaats bijvoorbeeld een openbare webserver in de DMZ.
- Een bastionhost kan in of nabij de DMZ staan, afhankelijk van de inrichting.



4. DMZ-verkeersregels

- Sta alleen benoemd, noodzakelijk verkeer toe, bijvoorbeeld internet → webserver in de DMZ.
- Sta beheer naar interne systemen alleen toe via het bastionhost-toegangspunt.
- Blokkeer overige verbindingen volgens de gekozen regels; controleer regels regelmatig.



5. Bastionhost

- Beperk toegang tot bevoegde beheerders.
- Gebruik de host als gecontroleerde route naar afgeschermde systemen.
- Leg toegang vast voor controle.

6. Beveiligingsmaatregelen

- Beperk toegestane verbindingen tussen internet, DMZ en intern netwerk met expliciete regels.
- Beveilig en controleer de bastionhost zorgvuldig.
- Houd er rekening mee dat een DMZ geen garantie tegen compromittering biedt.

7. Voorbeeld en aandachtspunten

- Voorbeeld: publiek bezoek bereikt de webserver in de DMZ.
- Beheerders bereiken interne systemen uitsluitend via het gecontroleerde bastionhost-toegangspunt.
- De DMZ en de bastionhost hebben verschillende functies; de precieze plaatsing hangt af van de architectuur.

8. Korte samenvatting

- DMZ = afgescheiden netwerksegment voor bereikbare diensten.
- Bastionhost = beveiligd en gecontroleerd toegangspunt voor beheer.
- Expliciete verkeersregels beperken verbindingen, maar nemen niet elk risico weg.

Web Application Firewalls (WAF)

1. Wat is het?

Een Web Application Firewall (WAF) bewaakt het webverkeer van en naar een webapplicatie en past regels toe op binnenkomende verzoeken.

- HTTP en HTTPS zijn afspraken voor communicatie tussen een browser en een webserver.
- Een verzoek kan bijvoorbeeld een pagina opvragen of gegevens naar een applicatie sturen.
- Een WAF kan een verzoek doorlaten, blokkeren of anders afhandelen.

2. Werking op laag 7

Een WAF werkt op de applicatielaag: laag 7 van het OSI-model. Daardoor kan hij kenmerken van webverzoeken bekijken. Een netwerkfirewall richt zich vooral op netwerk- en transportinformatie.

- Een WAF kan onder meer de methode, het pad, headers en parameters van een verzoek beoordelen.
- De plaatsing bepaalt welk verkeer de WAF kan inspecteren.

3. Signaturen en regels

Een regel beschrijft welk verzoek aandacht vraagt en wat ermee moet gebeuren. Signaturen zijn herkenningspatronen die kunnen passen bij bekende aanvalstechnieken.

- Regels kunnen kijken naar patronen en naar onderdelen zoals URL-pad of parameterwaarden.
- Een regel kan een verdacht verzoek markeren, blokkeren of registreren.
- Regels vragen onderhoud: veranderend verkeer kan aanpassing nodig maken.

4. Bescherming tegen webaanvallen

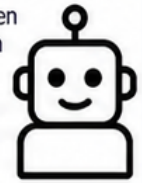
Een WAF kan helpen bij het herkennen of beperken van veelvoorkomende aanvallen op webapplicaties, zoals injectiepogingen en cross-site scripting (XSS).

- De werking hangt af van de regels, configuratie en het verkeer.
- Een WAF garandeert niet dat aanvallen worden gestopt.
- Veilig programmeren en het tijdig bijwerken van applicaties blijven noodzakelijk.

Aanvullende verdedigingslag
Gebruik een WAF naast veilige applicatieontwikkeling en onderhoud, niet als vervanging daarvan.

5. Geautomatiseerd verkeer

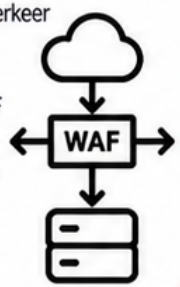
Niet al het geautomatiseerde verkeer is schadelijk: zoekmachines en andere legitieme diensten gebruiken ook bots. Herhaalde of afwijkende verzoeken kunnen aanleiding zijn voor extra controle.



- Limieten op het aantal verzoeken, gedragssignalen en reputatieregels kunnen helpen automatisering te herkennen of te beperken.
- Een te strenge regel kan ook legitieme bezoekers treffen; dat heet een false positive, oftewel een onterechte melding of blokkade.
- Regels moeten daarom worden afgestemd en gecontroleerd.

6. Plaatsing en verkeersstroom

Een WAF kan bijvoorbeeld als reverse proxy vóór de webapplicatie staan, of onderdeel zijn van een clouddienst. De gekozen plaatsing bepaalt hoe verkeer wordt gerouteerd en geïnspecteerd.



- Bij een reverse proxy loopt verkeer via de WAF voordat het de applicatie bereikt.
- Controleer welke verkeersroute daadwerkelijk door de WAF loopt.

7. Voorbeeld en aandachtspunten

Twee eenvoudige voorbeelden laten zien hoe regels kunnen reageren. Ze zijn illustratief: een regel is niet feilloos en moet passend worden ingesteld.

- **Verdachte parameter:** een regel markeert een ongebruikelijke invoer in een zoekparameter voor controle, zonder te veronderstellen dat iedere afwijkende invoer een aanval is.
- **Herhaald verkeer:** bij uitzonderlijk veel identieke verzoeken binnen korte tijd kan een ingestelde limiet verdere verzoeken tijdelijk afremmen.
- Controleer meldingen en mogelijke false positives; pas regels aan wanneer legitiem verkeer wordt geraakt.

8. Korte samenvatting

Een WAF inspecteert webverzoeken op applicatielaag 7 en gebruikt regels om verkeer te beoordelen.

- Signaturen en verzoekkenmerken kunnen helpen bekende patronen te herkennen.
- Limieten en gedragssignalen kunnen geautomatiseerd verkeer helpen beheersen.
- Een WAF is een aanvullende verdedigingslaag, geen volledige oplossing.

Honeypots & deceptiontechnologie

1. Wat is het?

- Een honeypot is een loksysteem of lokdienst die bewust aantrekkelijk lijkt voor ongewenste toegang.
- Het doel is activiteit te detecteren en te observeren.
- Een honeynet is een afgeschermde omgeving met meerdere samenwerkende loksystemen.

2. Hoe werkt het?

- Een lokmiddel wordt zo ingericht dat interactie ermee kan worden opgemerkt.
- Een melding kan aanleiding zijn voor nader onderzoek.
- Observaties worden vastgelegd en geanalyseerd.

3. Typen lokmiddelen

- Lokaccounts: accounts die uitsluitend bedoeld zijn om interactie te signaleren.
- Lokbestanden: bestanden die als lokmiddel dienen.
- Nagebootste diensten en schijnomgevingen: simulaties die activiteit kunnen aantrekken.

4. Wat kun je waarnemen?

- Welke lokmiddelen worden benaderd.
- Welk gedrag en welke technieken tijdens die interactie zichtbaar zijn.
- De logging kan observaties opleveren voor analyse, maar toont niet alles.

5. Toepassingen

- Vroegtijdig signaleren van activiteit rond lokmiddelen.
- Observaties verzamelen voor verdere analyse.
- Detectieprocessen toetsen en nagaan of meldingen worden opgevolgd.

6. Veilige inrichting

- Bepaal vooraf het doel en de grenzen van het loksysteem.
- Isoleer het zorgvuldig, zodat activiteit andere systemen niet bereikt.
- Beperk toegang en zorg voor logging en bewaking.
- Behandel een melding als aanleiding voor onderzoek, niet als sluitend bewijs.

Veiligheidsbasis

Een loksysteem moet afgeschermd en bewaakt zijn. Het vervangt geen reguliere beveiligingsmaatregelen.

7. Beperkingen

- Een honeypot geeft geen volledig beeld van alle aanvallers of aanvallen.
- Geen interactie waarnemen betekent niet dat er geen ongewenste activiteit is.
- Waarnemingen vragen om context en nader onderzoek.
- Gebruik de uitkomsten naast reguliere beveiligingsmaatregelen.

8. Korte samenvatting

- Honeypots en andere lokmiddelen kunnen activiteit zichtbaar maken.
- Een honeynet groepeert meerdere loksystemen in een afgeschermde omgeving.
- Veilige isolatie, beperkte toegang, logging, bewaking en een duidelijk doel zijn belangrijk.
- De observaties ondersteunen analyse, maar bieden geen volledige zichtbaarheid.

Netwerktoegangsbeheer (NAC) & 802.1X

1 Wat is het?

- NAC: beleid bepaalt welke apparaten en gebruikers netwerktoegang krijgen.
- De beslissing kan afhangen van identiteit en apparaatstatus.

2 Waarom gebruiken?

- Beheer toegang op basis van regels, niet alleen op basis van een netwerkverbinding.
- Weeg naast identiteit ook kenmerken of naleving van het apparaat mee.
- Mogelijke uitkomsten: toegang toestaan, beperken of weigeren.

3 Betrokken rollen

- **Supplicant:** apparaat of software die toegang aanvraagt.
- **Authenticator:** bijvoorbeeld een switch of draadloos toegangspunt; bemiddelt bij de toegang.
- **Authenticatieserver:** beoordeelt de aanmelding, bijvoorbeeld met RADIUS.

4 802.1X en poortcontrole

- IEEE 802.1X biedt poortgebaseerde netwerktoegangscontrole.
- De supplicant vraagt toegang aan; de authenticator bemiddelt.
- De authenticatieserver beoordeelt de aanmelding.

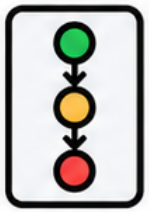
5 EAP als raamwerk

- EAP is het raamwerk voor het uitwisselen van authenticatieberichten.
- Het ondersteunt verschillende authenticatiemethoden.
- EAP is zelf geen afzonderlijke authenticatiemethode.

6 RADIUS-server

- RADIUS kan dienen als centrale authenticatie- en autorisatieserver.
- De authenticator stuurt authenticatieverkeer door naar RADIUS.
- Het antwoord kan toegang toestaan of weigeren en toegangskenmerken bevatten.

7 Verificatiestroom en uitkomsten



- Apparaat verbindt → 802.1X/EAP verloopt via de authenticator → RADIUS beoordeelt de gegevens.
- Het NAC-beleid bepaalt vervolgens de toegang.
- Bij beperkte toegang kan een herstel- of quarantainenetwerk worden gebruikt.

8 Korte samenvatting

- **NAC:** het bredere, beleidsgestuurde toegangsbeheer.
- **802.1X:** poortgebaseerde toegangscontrole.
- **EAP:** raamwerk voor authenticatie-uitwisseling.
- **RADIUS:** mogelijke centrale server voor authenticatie en autorisatie.

Data Loss Prevention (DLP) & gegevensexfiltratie

1. Wat is het

- Beschermt gevoelige gegevens tegen ongeoorloofde blootstelling of overdracht.
- Combineert beleid met technische controles.
- Herkennen, beperken en passend reageren.

2. Gegevensfasen

- In rust: opgeslagen, bijvoorbeeld in cloudopslag.
- In beweging: verzonden via netwerk, e-mail of ander kanaal.
- In gebruik: verwerkt of weergegeven door gebruiker of toepassing.

3. Gevoelige gegevens

- Persoonsgegevens.
- Vertrouwelijke documenten, zoals interne rapporten.
- Betaalkaartgegevens en andere informatie met beperkte toegang.

4. Detectiemethoden

- Patroonregels zoeken herkenbare gegevensvormen.
- Classificatielabels bepalen behandeling.
- Contextuele kenmerken helpen beoordelen of gebruik of overdracht past.

5. Bewaken en blokkeren

- Bewaak e-mail, webverkeer, cloudopslag, eindpunten en verwisselbare media.
- Mogelijke acties: waarschuwen, blokkeren, versleuteling afdwingen of overdracht vasthouden voor beoordeling.
- Kies op basis van classificatie en context.

6. Gegevensexfiltratie herkennen

- Ongeoorloofd wegsluizen van gegevens.
- Mogelijke signalen zijn ongebruikelijke overdracht, afwijkende bestemming, groot volume of opvallend gebruik van opslagmedia.
- Een signaal op zichzelf bewijst geen kwaadwiligheid.

7. Praktijk en aandachtspunten

- Voorbeeld: een e-mailbijlage met vertrouwelijke gegevens wordt herkend aan label en bestemming.
- Koppel classificatie en context aan een proportionele actie.
- Beoordeel legitieme uitzonderingen en fout-positieven.
- DLP is één beschermingslaag en vervangt geen toegangsbeheer, versleuteling, logging, incidentrespons of bewustwording.

8. Korte samenvatting

- Bescherm gegevens in rust, beweging en gebruik.
- Combineer inhoud, classificatie en context.
- Beleid bepaalt de reactie.
- Waarschuwing is geen bewijs van exfiltratie.

Endpointdetectie en respons (EDR) & uitgebreide detectie en respons (XDR)

1 Wat is EDR?

EDR bewaakt eindapparaten zoals laptops en servers. Het verzamelt en analyseert beveiligingsgegevens om verdachte gebeurtenissen te helpen herkennen en onderzoeken.

- Endpoint = eindapparaat, zoals een laptop of server.
- Detection and Response = detectie en respons.
- EDR ondersteunt beveiligingsteams; het vervangt geen beoordeling.

2 Telemetrie verzamelen.

Telemetrie bestaat uit gegevens over activiteiten op een eindapparaat. EDR kan zulke signalen samenbrengen voor analyse.

- Processtarts: welke programma's beginnen te draaien.
- Bestandswijzigingen: welke bestanden worden aangepast.
- Netwerkverbindingen: met welke systemen het appraaat communiceert.

3 Gedragsanalyse.

Gedragsanalyse beoordeelt activiteiten in hun context. Zo kunnen ook patronen opvallen die niet alleen op bekende schadelijke bestanden wijzen.

- Een enkele gebeurtenis vertelt niet altijd het hele verhaal.
- Combinaties en volgorde van gebeurtenissen kunnen belangrijk zijn.
- Een detectie kan een fout-positief zijn en vraagt om beoordeling.

4 Detecties en waarschuwingen.

Wanneer signalen op mogelijk verdacht gedrag wijzen, kan EDR een detectie of waarschuwing tonen. Die is een aanleiding voor onderzoek, niet automatisch bewijs van een incident.

- Bekijk welke gebeurtenissen de waarschuwing ondersteunen.
- Controleer tijdstip, apparaat en samenhangende activiteiten.
- Leg bevindingen vast volgens de werkwijze van de organisatie.

Let op: Een waarschuwing kan onschuldig gedrag betreffen. **Beoordeel de context voordat je conclusies trekt.**

5 Responsacties.

Afhankelijk van de oplossing, configuratie en het beleid kan EDR acties aanbieden om een incident te beheersen.

- Een eindapparaat isoleren van het netwerk.
- Een proces stoppen.
- Acties kunnen de beschikbaarheid beïnvloeden en horen bij een beheerde responsprocedure.

6 Onderzoek en context.

Onderzoek brengt relevante signalen samen om te begrijpen wat er mogelijk is gebeurd. Een waarschuwing kan daarbij richting geven, maar moet worden getoetst aan de beschikbare informatie.

- Voorbeeld: een ongebruikelijke processtart op een laptop, gevolgd door verdachte netwerkcommunicatie.
- Onderzoek de gebeurtenissen in hun tijdsvolgorde en samenhang.
- Dit voorbeeld is illustratief; niet elk product detecteert dit patroon.

7 Van EDR naar XDR.

XDR staat voor Extended Detection and Response. Het breidt detectie en respons uit over meerdere beveiligingsdomeinen of databronnen, zodat gebeurtenissen kunnen worden samengebracht en met elkaar in verband gebracht.

- Mogelijke bronnen zijn endpoints, e-mail en netwerk.
- De precieze dekking verschilt per oplossing.
- XDR bouwt voort op signalen uit meerdere domeinen; het is geen garantie dat elk verband wordt gevonden.

8 Korte samenvatting.

EDR verzamelt en analyseert beveiligingsrelevante telemetrie van eindapparaten. Gedragsanalyse kan verdachte patronen signaleren; waarschuwingen vragen om beoordeling. Afhankelijk van configuratie en beleid kan een responsactie worden ingezet. XDR brengt signalen uit meerdere beveiligingsdomeinen samen.

- EDR: bewaken, signaleren en onderzoek ondersteunen.
- Context helpt waarschuwingen te beoordelen; fout-positieven zijn mogelijk.
- XDR verbindt informatie uit meerdere bronnen; dekking verschilt per oplossing.

Identiteits- en toegangsbeheer (IAM)

IAM beheert digitale identiteiten en bepaalt wie toegang heeft tot systemen en gegevens.

1. Wat is IAM?

- Beheer van digitale identiteiten en toegangsrechten.
- Verbindt gebruikers met de systemen en gegevens die zij nodig hebben.

2. Gebruikerslevenscyclus

- **Aanmaken:** registreer een account wanneer toegang nodig is.
- **Wijzigen:** werk gegevens en rechten bij als de rol verandert.
- **Beoordelen:** controleer periodiek of toegang nog nodig is.
- **Intrekken:** verwijder of schakel toegang uit wanneer die niet meer nodig is.

3. Directoryservices

- Bewaren identiteits- en organisatiegegevens op een centrale plek.
- Maken het mogelijk gegevens centraal op te zoeken en te beheren.
- Voorbeeld: gebruikersgegevens en teaminformatie raadplegen in een bedrijfsdirectory.

4. Authenticatie

- Controleert wie een gebruiker is.
- Gebruikt factoren zoals kennis (wachtwoord), bezit (beveiligingssleutel) of een inherente eigenschap.
- Multifactorauthenticatie combineert factoren uit verschillende categorieën.

5. Autorisatie

- Bepaalt welke handelingen of bronnen een gebruiker mag gebruiken.
- Least privilege: geef alleen de minimale rechten die nodig zijn voor de taak.
- Voorbeeld: een medewerker mag een bestand lezen, maar niet wijzigen.

6. Toegangsmodellen: RBAC en ABAC

- **RBAC** — rolgebaseerde toegangscontrole: rechten zijn gekoppeld aan rollen, zoals 'beheerder' of 'lezer'.
- **ABAC** — attribuutgebaseerde toegangscontrole: beleid weegt kenmerken van gebruiker, bron en context mee.
- Beide modellen helpen bepalen welke toegang passend is.

7. Centrale governance

- Leg beleid en eigenaarschap voor toegang vast.
- Beoordeel rechten om te controleren of ze nog nodig zijn.
- Registreer relevante toegangsgebeurtenissen.

8. Korte samenvatting

- IAM beheert identiteiten en toegang.
- Authenticatie controleert wie iemand is; autorisatie bepaalt wat diegene mag doen.
- Beheer accounts gedurende hun levenscyclus, pas minimale rechten toe en beoordeel toegang periodiek.

RBAC & ABAC

1. Wat is het

- Bepaalt toegang ná aanmelding
- Gaat over wie, wat en welke handeling
- Voorbeeld: een gebruiker mag een document bekijken, maar niet wijzigen

2. RBAC-rollen

- Lezer: documenten bekijken
- Beheerder: gebruikers en instellingen beheren
- Geef alleen rollen die bij de taak passen

3. ABAC-attributen

- Gebruiker: afdeling
- Object: documentclassificatie
- Context: locatie of tijdstip
- Voorbeeld: een medewerker kan een document alleen bekijken als afdeling en classificatie aan de voorwaarden voldoen

4. Beleidsregels

- Regels kunnen meerdere attributen combineren
- Een regel kan per verzoek opnieuw worden beoordeeld
- Voorbeeld: toegang toestaan als de afdeling klopt én het document voor die afdeling toegankelijk is

5. Toegangsbeoordeling

- Wie vraagt toegang?
- Welk object en welke handeling?
- Welke rol, attributen en context zijn van toepassing?
- Uitkomst: toestaan of weigeren

6. RBAC en ABAC vergeleken

- RBAC: bevoegdheden rond rollen
- ABAC: voorwaarden op basis van attributen
- ABAC kan fijnmazig rekening houden met context
- Een systeem kan beide modellen naast elkaar gebruiken

7. Minimale privileges

- Beperk rechten passend bij de taak
- Herzie rechten wanneer taken veranderen
- Voorbeeld: een lezer krijgt geen beheerrechten

8. Korte samenvatting

- RBAC: wie heeft welke rol?
- ABAC: voldoen kenmerken en context aan de regel?
- Minimale privileges: niet meer rechten dan nodig

Meervoudige verificatie (MFA) en FIDO2

1. Wat is het

Meervoudige verificatie (MFA) controleert je identiteit met minstens twee verschillende soorten bewijs. Zo biedt aanmelden meer bescherming dan alleen een wachtwoord.

- Twee wachtwoorden tellen als één soort factor.
- MFA is sterker dan één wachtwoord, maar de bescherming hangt af van de gebruikte factoren en aanmeldroute.

2. Authenticatiefactoren

Een factor is een soort bewijs waarmee je laat zien dat je bent wie je zegt te zijn.

- Iets wat je weet: bijvoorbeeld een wachtwoord of pincode.
- Iets wat je hebt: bijvoorbeeld een telefoon of beveiligingssleutel.
- Iets wat je bent: bijvoorbeeld een biometrisch kenmerk.
- MFA combineert minstens twee verschillende factorsoorten.

3. MFA in de praktijk

Bij aanmelden controleert een dienst meerdere bewijzen, bijvoorbeeld een wachtwoord én een code of beveiligingssleutel.

- Voorbeeld: wachtwoord (weten) + code op je telefoon (hebben).
- Twee wachtwoorden combineren niet twee factorsoorten.
- De gekozen methode beïnvloedt hoe goed MFA bestand is tegen phishing.

8. Korte samenvatting

MFA combineert verschillende factorsoorten; FIDO2/WebAuthn biedt verificatie die aan de website-oorsprong is gebonden.

- Weten, hebben en zijn zijn de drie factorsoorten.
- Sms en TOTP voegen een stap toe, maar zijn niet phishingbestendig.
- Bij FIDO2 blijft de privésleutel bij de authenticator.
- Passkeys maken FIDO-aanmelden gebruiksvriendelijk; controleer ook welke herstelroute bestaat.

4. Sms-codes

Een dienst kan een tijdelijke code per sms sturen als extra aanmeldstap.

- De code kan worden onderschept of via phishing worden buitgemaakt.
- Misbruik van een overgezet telefoonnummer kan sms-verificatie omzeilen.
- Een sms-code is nuttiger dan alleen een wachtwoord, maar niet phishingbestendig.

7. Wachtwoordloos aanmelden

Wachtwoordloos aanmelden gebruikt geen traditioneel wachtwoord als primaire factor. Passkeys zijn een gebruiksvriendelijke vorm van FIDO-verificatie.

- Afhankelijk van de implementatie kunnen passkeys tussen apparaten synchroniseren of aan één beveiligingssleutel gekoppeld zijn.
- Wachtwoordloos betekent niet automatisch dat er geen herstel- of alternatieve aanmeldroute is.
- Herstelmethoden zijn daarom ook onderdeel van het aanmeldproces.

6. FIDO2 en WebAuthn

FIDO2 ondersteunt aanmelden met een authenticator, zoals een beveiligingssleutel of een apparaat. WebAuthn regelt de communicatie tussen website en authenticator; CTAP die tussen authenticator en apparaat.

- Bij registratie ontstaat een sleutelpaar: de privésleutel blijft bij de authenticator; de website bewaart de publieke sleutel.
- Bij aanmelden ondertekent de authenticator een uitdaging. De privésleutel wordt niet naar de website gestuurd.
- WebAuthn bindt verificatie aan de website-oorsprong. Een nagemaakt domein krijgt daardoor niet dezelfde verificatie.

5. TOTP-codes

Een authenticator-app kan een eenmalige tijdscode (TOTP) lokaal genereren. De app gebruikt daarvoor een gedeeld geheim en een tijdsinterval.

- De code verandert na een korte periode.
- Phishing kan de code buitmaken; een aanvaller kan die direct proberen te gebruiken.
- TOTP is nuttiger dan alleen een wachtwoord, maar niet phishingbestendig.

Bevoegd toegangsbeheer (PAM)

1. Wat is PAM?

- Beheert accounts en rechten met verhoogde bevoegdheden.
- Voorbeelden: systeembeheerder- en serviceaccounts.

2. Waarom is PAM belangrijk?

- Helpt onnodige of permanente beheerrechten te beperken.
- PAM-maatregelen helpen risico's verkleinen, maar bieden geen absolute beveiliging.

3. Beheerkluis

- Beheert bevoorrechte accounts en geheimen, zoals wachtwoorden, centraal.
- Beperk toegang tot de kluis tot bevoegde gebruikers.

4. Sessiemonitoring

- Registreert bevoorrechte sessies of volgt ze in real time.
- Maakt toezicht en latere beoordeling mogelijk.

5. Just-in-Time-toegang

- Verleent verhoogde rechten tijdelijk, wanneer ze nodig zijn.
- Voorbeeld: een beheerder krijgt voor onderhoud kort toegang tot een server.

6. Rotatie van inloggegevens

- Vervangt wachtwoorden of andere geheimen periodiek of na gebruik.
- Helpt langdurig hergebruik te beperken.

7. Praktijkvoorbeeld en aandachtspunten

- Een beheerder vraagt tijdelijk rechten aan voor een goedgekeurde serverupdate.
- Gebruik waar mogelijk persoonlijke accounts en deel geen wachtwoorden.
- Leg vast wie toegang kreeg en controleer sessies volgens het beleid.

Let op
Een beheerkluis of monitoring vervangt zorgvuldig rechtenbeheer en toezicht.

8. Korte samenvatting

- PAM beheert accounts en verhoogde toegangsrechten.
- Een beheerkluis, sessiemonitoring, tijdelijke toegang en rotatie helpen risico's beperken.
- Ken alleen noodzakelijke rechten toe en houd toezicht.

047 · Single Sign-On (SSO) & federatieve identiteit

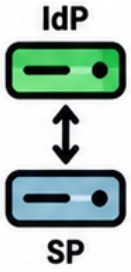
1. Wat is SSO?

Single Sign-On (SSO) laat je na één aanmelding meerdere gekoppelde diensten gebruiken zonder bij elke dienst opnieuw in te loggen.

- Eén aanmelding voor meerdere diensten.
- Elke dienst bepaalt zelf welke toegangsrechten gelden.

2. IdP en SP

Bij SSO werken een identiteitsprovider en een dienst samen.



- IdP (identiteitsprovider): authenticeert de gebruiker.
- SP (serviceprovider): vraagt toegang aan en beoordeelt de ontvangen verklaring of het token.
- De SP bepaalt welke rechten de gebruiker krijgt.

3. Federatieve identiteit

Bij federatieve identiteit vertrouwen afzonderlijke organisaties of diensten op afgesproken identiteitsgegevens en protocollen.

- De IdP bevestigt de identiteit.
- De SP gebruikt de afgesproken gegevens om toegang te beoordelen.
- Voorbeeld: aanmelden bij een externe dienst met een werkaccount.

4. SAML 2.0

SAML 2.0 is een op XML gebaseerde standaard voor het uitwisselen van authenticatie- en autorisatiegegevens tussen partijen.

- Wordt gebruikt om informatie over een aanmelding tussen een IdP en een SP uit te wisselen.
- Een SAML-assertion kan gegevens bevatten over authenticatie, het subject en attributen.
- De SP controleert de assertion en de geldigheidsvoorwaarden.

5. Assertion en token

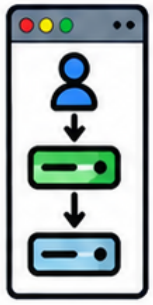
Een assertion of token is een digitale verklaring die informatie over een gebruiker of aanmelding kan bevatten. Het is geen wachtwoord.

- Deel of log tokens en assertions niet onnodig.
- Controleer onder meer handtekening, uitgever, ontvanger en geldigheidsperiode.
- Beperk blootstelling en bewaartermijn.

Let op: Een onderschept of gelekt token kan toegang mogelijk maken. Behandel het daarom als gevoelige informatie.

6. Inlogstroom

Een veelvoorkomende SAML-aanmelding via de browser verloopt als volgt:



1. **Vraag toegang aan** — Je opent een beschermde dienst (SP).
2. **Ga naar de IdP** — De SP leidt je browser door naar de identiteitsprovider.
3. **Authenticeer** — De IdP controleert je aanmelding en maakt een assertion.
4. **Beoordeel de assertion** — De assertion wordt via de browser bij de SP bezorgd. De SP controleert de verklaring voordat toegang wordt verleend.

7. Beveiliging en aandachtspunten

SSO maakt aanmelden en centraal identiteitsbeheer eenvoudiger, maar kan ook risico's concentreren: een gecompromitteerd account of een getroffen IdP kan meerdere gekoppelde diensten raken.

- Gebruik waar passend multifactorauthenticatie (MFA).
- Geef alleen de minimaal benodigde toegangsrechten.
- Beheer sessies zorgvuldig en leg relevante gebeurtenissen veilig vast.
- Valideer assertions en bescherm de verbinding tijdens transport.

Onthoud: Gemak door SSO vervangt niet de toegangscontrole van elke afzonderlijke dienst.

8. Korte samenvatting

SSO laat je met één aanmelding meerdere gekoppelde diensten gebruiken. Federatieve identiteit maakt afspraken tussen partijen mogelijk; de IdP authenticeert en de SP beoordeelt de ontvangen assertion of het token.

- SAML 2.0 is een XML-gebaseerde uitwisselingsstandaard.
- Een assertion is geen wachtwoord: valideer en bescherm deze.
- MFA, minimale rechten en zorgvuldig sessiebeheer helpen risico's beperken.

Mindmap 049 — Active Directory-beveiliging: structuur en Kerberos.

1. Wat is Active Directory?

- Directorydienst voor het organiseren en beheren van netwerkbronnen.
- Beheert onder meer gebruikers, computers en groepen.

2. Forest en domeinen

- Een domein groepeert directory-objecten.
- Domeinen kunnen samen een forest vormen.
- Een forest omvat een gemeenschappelijke directory- en vertrouwensgrens.

3. Identiteit en accounts

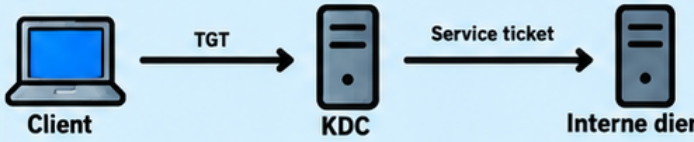
- Gebruikersaccounts vertegenwoordigen personen of services.
- Computeraccounts vertegenwoordigen apparaten in het domein.
- Groepen bundelen accounts voor beheer en toegangsverlening.

4. Kerberos in hoofdlijnen

- Kerberos is een protocol voor netwerkverificatie.
- De **client vraagt** tickets aan bij het Key Distribution Center (KDC).
- Tickets helpen de identiteit te verifiëren zonder het wachtwoord telkens aan een dienst te sturen.

5. Tickets: TGT en service ticket

- Eerst vraagt de client bij het KDC een ticket-granting ticket (TGT) aan.
- Met het TGT vraagt de client een service ticket (TGS-ticket) aan voor een specifieke dienst.
- De client presenteert het service ticket aan die dienst.



Voorbeeld: Na aanmelding gebruikt een client het TGT om een ticket voor een interne dienst aan te vragen.

6. SPN's

- Een Service Principal Name (SPN) koppelt een dienst aan het account waaronder die dienst draait.
- Kerberos gebruikt SPN's om het juiste dienstaccount te vinden.
- Unieke, correct geregistreerde SPN's ondersteunen betrouwbare authenticatie.

7. Basale hardening

- Pas least privilege toe: geef alleen benodigde rechten.
- Beperk bevoorrechte accounts en controleer ze periodiek.
- Gebruik sterke, unieke wachtwoorden en installeer beveiligingsupdates tijdig.
- Beperk onnodige diensten en monitor relevante aanmeldingen en directorywijzigingen.

8. Korte samenvatting

- Domeinen organiseren objecten; een forest groepeert domeinen.
- Kerberos-ticketstroom: client → KDC (TGT) → KDC (service ticket) → dienst.
- SPN's koppelen diensten aan dienstaccounts.
- Geef accounts niet meer rechten dan nodig en houd beheer en monitoring bij.

Active Directory-aanvallen

1 Wat is het?

Active Directory (AD) beheert gebruikers, computers en toegangsrechten in veel Windows-domeinen. Aanvallers kunnen misbruik maken van zwakke accountbeveiliging of gestolen aanmeldgegevens. Herkenning en bescherming staan centraal.

- Kerberos regelt ticketgebaseerde aanmelding in een domein.
- Een ticket of hash is geen gewoon wachtwoord, maar kan wel misbruikt worden bij authenticatie.

2 Kerberoasting

Bij Kerberoasting kunnen aanvallers met domeintoegang service tickets aanvragen voor accounts die aan een Service Principal Name (SPN) zijn gekoppeld. Gegevens uit die tickets kunnen vervolgens offline worden onderzocht op zwakke wachtwoorden.

- Een SPN koppelt een dienst aan een account.
- Zwakke serviceaccountwachtwoorden verhogen het risico.
- Voorbeeld: een verouderd serviceaccount met een eenvoudig wachtwoord is kwetsbaarder.

3 AS-REP Roasting

Deze techniek richt zich op accounts waarvoor Kerberos-pre-authenticatie is uitgeschakeld. Daardoor kan een aanvaller ticketgegevens verkrijgen en offline wachtwoordgissingen uitvoeren.

- Pre-authenticatie controleert vooraf of een gebruiker zich mag aanmelden.
- Schakel deze controle alleen uit als dat noodzakelijk is en de risico's zijn afgewogen.

4 Pass-the-Hash

Bij Pass-the-Hash gebruikt een aanvaller een verkregen wachtwoordhash om zich bij bepaalde Windows-netwerkdiensten te authenticeren, zonder eerst het oorspronkelijke wachtwoord te herstellen.

- De hash wordt hergebruikt voor authenticatie.
- Dit is iets anders dan een plaintext-wachtwoord hergebruiken: dat is het leesbare wachtwoord zelf.

5 Golden Ticket

Een Golden Ticket is een vervalst Kerberos-ticket voor domeinaanmelding (TGT). Het maken ervan kan mogelijk worden na compromittering van het geheim van het KRBTGT-account; het kan ongeautoriseerde toegang ondersteunen zolang het ticket wordt geaccepteerd.

- Het KRBTGT-account speelt een rol bij het ondertekenen van TGT's.
- Bescherm de bijbehorende referenties en beperk toegang tot beheerdersaccounts.

6 Risico's en signalen

Deze technieken verschillen in wat wordt misbruikt. Let bij onderzoek op ongebruikelijke ticketaanvragen en aanmeldingen en volg de procedures van je organisatie.

- Kerberoasting en AS-REP Roasting: kunnen offline wachtwoordgissingen mogelijk maken.
- Pass-the-Hash: hergebruikt een hash voor authenticatie.
- Golden Ticket: misbruikt het geheim waarmee TGT's worden ondertekend.
- Een afwijkende aanmelding of ticketaanvraag is een signaal om nader te onderzoeken, geen bewijs op zichzelf.

7 Mitigaties

Verklein de kans op misbruik door accounts, referenties en toegangsrechten zorgvuldig te beheren. Stem wijzigingen af op de functie van diensten en op het beveiligingsbeleid van je organisatie.

- Gebruik sterke, unieke en beheerde wachtwoorden voor serviceaccounts.
- Beperk onnodige SPN's en serviceaccounts.
- Schakel pre-authenticatie alleen uit wanneer dat noodzakelijk is.
- Beperk beheerdersrechten en bescherm aanmeldgegevens.
- Gebruik Credential Guard waar passend.
- Monitor ongebruikelijke ticketaanvragen en aanmeldingen.

8 Korte samenvatting

Denk aan het verschil tussen ticketmisbruik, hashhergebruik en zwakke wachtwoorden. Goede accountbeveiliging, beperkte rechten en monitoring helpen risico's te verkleinen.

- Kerberoasting: service tickets en mogelijk zwakke serviceaccountwachtwoorden.
- AS-REP Roasting: accounts zonder Kerberos-pre-authenticatie.
- Pass-the-Hash: een hash gebruiken voor authenticatie.
- Golden Ticket: een vervalst TGT, mogelijk na compromittering van het KRBTGT-geheim.
- Verdediging: beheer accounts en rechten zorgvuldig en onderzoek afwijkende signalen.

Windows-systeemverharding en groepsbeleid

1. Wat is systeemverharding?

Verklein het aanvalsoppervlak door Windows-configuraties te beveiligen en onnodige functies te beperken.

- Beperk wat niet nodig is.
- Stem beveiliging af op de omgeving.

2. GPO's en baselines

Groepsbeleidsobjecten (GPO's) passen instellingen centraal toe op gebruikers en computers. Een baseline is een samenhangende uitgangsset van beveiligingsinstellingen.

- Begin met een passende baseline.
- Toets wijzigingen aan de behoeften en compatibiliteit van de omgeving.
- Controleer na toepassing of de bedoelde beleidsinstellingen zijn aangekomen.

3. Services beperken

Schakel een service alleen uit als die niet nodig is. Een ondoordachte wijziging kan afhankelijkheden of bedrijfsfuncties verstoren.

- Controleer het doel en de afhankelijkheden.
- Test eerst op een beperkte groep.
- Controleer na toepassing of essentiële functies blijven werken.

4. BitLocker

BitLocker versleutelt volumes en helpt gegevens beschermen als een apparaat verloren raakt of wordt gestolen.

- Beheer sleutels en herstelgegevens zorgvuldig.
- Beperk toegang tot herstelsleutels tot bevoegde beheerders.
- Controleer vooraf of hersteltoegang beschikbaar is voor een bevoegde beheerder.

5. Defender-mitigaties

Microsoft Defender-beveiligingsfuncties kunnen bijdragen aan het voorkomen en detecteren van dreigingen.

- Houd beveiligingsbescherming ingeschakeld.
- Gebruik actuele beveiligingsinformatie.
- Beperk uitzonderingen en beheer ze zorgvuldig.

6. Beheer en herstel

Voer configuratiewijzigingen gecontroleerd uit, zodat beheerders afwijkingen kunnen herkennen en herstellen.

- Leg wijzigingen en redenen vast.
- Houd herstelgegevens veilig beschikbaar.
- Controleer de beleidsresultaten na toepassing.
- Herstel de vorige instelling als essentiële functies uitvallen.

7. Praktijkvoorbeeld: ongebruikte service via GPO beperken

Een beheerteam wil een service die niet nodig is op beheerde computers beperken. De wijziging wordt eerst gecontroleerd en kleinschalig getest.

- Controleer het gebruik, de bedrijfsfunctie en afhankelijkheden van de service.
- Pas de wijziging via een passende GPO toe op een beperkte testgroep.
- Controleer of de beleidsinstelling is aangekomen en of benodigde functies blijven werken.
- Rol de wijziging terug als een essentiële functie uitvalt.

8. Korte samenvatting

Verhard Windows stapsgewijs: gebruik passende GPO-baselines, beperk alleen onnodige services, bescherm volumes met BitLocker en beheer Defender-instellingen zorgvuldig.

- Test wijzigingen en controleer het resultaat.
- Beheer hersteltoegang zorgvuldig.
- Pas beveiliging aan de omgeving aan.

Windows Event Logging & Sysmon-bewaking

Windows-auditing en Sysmon bieden aanvullende aanwijzingen over systeemactiviteit. Beoordeel gebeurtenissen altijd in context.

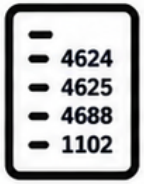
1. Wat is het

Gebeurtenisregistratie legt systeemactiviteit vast, zodat beheerders die achteraf kunnen onderzoeken.

- Windows-auditing registreert beveiligingsgebeurtenissen.
- Sysmon vult Windows-auditing aan met extra systeemdetails.
- Logboeken helpen patronen te onderzoeken; één gebeurtenis bewijst op zichzelf geen incident.

Onthoud: Logs zijn aanwijzingen. Beoordeel ze altijd in context.

2. Security-logboek en Event ID's



Bekijk Windows-beveiligingsgebeurtenissen in Logboeken via Windows-logboeken > Beveiliging.

- 4624 — geslaagde aanmelding.
- 4625 — mislukte aanmelding.
- 4688 — nieuw proces gemaakt, als procescontrole is ingeschakeld.
- 1102 — auditlogboek gewist.

Beschikbaarheid en betekenis zijn afhankelijk van auditbeleid en Windows-configuratie.

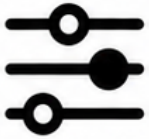
3. Sysmon: rol en installatie

Sysmon is een Windows-systeemservice met stuurprogramma die systeemactiviteit kan vastleggen in Microsoft-Windows-Sysmon/Operational.

- Gebruik Sysmon naast Windows Security-auditing, niet als vervanging.
- Demonstreer alleen op een testcomputer met een gecontroleerde configuratie.
- Volg voor installatie de officiële documentatie; test wijzigingen eerst buiten productie.
- Genereer onschadelijke testactiviteit en bekijk de gebeurtenissen in Logboeken.

Veilig oefenen: Voer configuratie- en installatietests uit in een niet-productieomgeving.

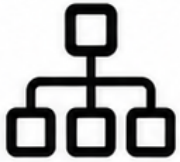
4. Configuratie en gebeurtenisbronnen



De configuratie bepaalt welke gebeurtenissen Sysmon registreert. Dezelfde gebeurtenis is niet in iedere omgeving beschikbaar.

- Procescreatie — Sysmon-gebeurtenis 1.
- Netwerkverbinding — gebeurtenis 3, indien ingeschakeld.
- Bestandscreatie — gebeurtenis 11.
- Een te beperkte configuratie kan blinde vlekken geven; een te ruime configuratie kan veel ruis opleveren.
- Windows Security en Sysmon hebben verschillende logboeken en kunnen elkaar aanvullen.

5. Procescreatie lezen



Een procescreatiegebeurtenis kan details tonen die helpen om te begrijpen wat er is gestart en vanuit welke context.

- Bekijk de procesnaam en het uitvoerbare pad.
- Controleer het bovenliggende proces en de gebruikerscontext.
- Lees de opdrachtregel zorgvuldig; bron en configuratie bepalen welke velden beschikbaar zijn.
- Vergelijk met normaal gebruik op het apparaat voordat je conclusies trekt.

Voorbeeld: Een onverwachte opdrachtregel kan aanleiding zijn voor nader onderzoek, maar is op zichzelf geen bewijs van kwaadwilligheid.

6. Verdachte signalen herkennen

Let op afwijkingen en onderzoek ze met aanvullende gegevens.

- Een programma start vanuit een ongebruikelijke schrijfbare map.
- Een onbekend proces heeft een onverwacht bovenliggend proces.
- Meerdere mislukte aanmeldingen worden gevolgd door een geslaagde aanmelding.
- Een signaal is een aanwijzing, geen definitief oordeel.

Voorzichtig interpreteren: Een afwijking kan ook een legitieme verklaring hebben. Onderzoek eerst de context.

7. Onderzoek en correlatie

Combineer gebeurtenissen en context om een tijdlijn op te bouwen en de betekenis van een signaal te beoordelen.

- Vergelijk tijdstippen, accounts, processen en apparaten.
- Zoek naar samenhangende gebeurtenissen in Security- en Sysmon-logboeken.
- Controleer of het gedrag past bij bekende, toegestane activiteiten.
- Leg vast wat je waarnam en welke onzekerheden overblijven.
- Behandel een gebeurtenis pas als incident na beoordeling van het geheel.

8. Korte samenvatting

Windows-auditing en Sysmon bieden verschillende, aanvullende inzichten in systeemactiviteit.

- 4624: geslaagde aanmelding; 4625: mislukte aanmelding.
- Sysmon-gebeurtenis 1 kan procescreatie vastleggen; beschikbare details hangen af van configuratie.
- Gebruik gecontroleerde configuraties en oefen buiten productie.
- Correlatie en context maken van ruwe logs bruikbare aanwijzingen.

Linux-beveiligings-fundamenten: machtigingen en sudo

1. Wat is het

Linux-machtigingen bepalen wie bestanden en mappen mag gebruiken.

- Drie categorieën: eigenaar, groep en overige gebruikers.
- Doel: toegang beperken tot wat nodig is.

2. POSIX-rechten rwx

De letters geven aan welke handelingen zijn toegestaan.

- r — lezen; bij een map: namen bekijken.
- w — schrijven; bij een map: inhoud aanmaken, wijzigen of verwijderen.
- x — uitvoeren; bij een map: toegang tot inhoud via die map.

3. Rechten lezen en noteren

Een rechtenreeks toont achtereenvolgens eigenaar, groep en overige gebruikers.

- rwxr-xr-- = eigenaar rwx, groep r-x, overige r--.
- Numeriek: r=4, w=2, x=1; tel per categorie op.
- 755 = rwxr-xr-x: eigenaar alles, groep en overige lezen en uitvoeren.

4. SUID en SGID

SUID en SGID zijn speciale permissiebits; hun werking hangt af van bestandstype en configuratie.

- Op uitvoerbare bestanden kunnen ze de effectieve gebruikers- of groepsidentiteit beïnvloeden.
- Op mappen zorgt SGID op gangbare Unix-achtige systemen doorgaans dat nieuwe items de mapgroep erven.
- Onnodige of verkeerd ingestelde bits kunnen risico's geven; beoordeel ze per geval.

5. sudo en sudoers

sudo laat bevoegde gebruikers opdrachten volgens beleid met een andere identiteit uitvoeren, vaak die van root.

- Ken alleen noodzakelijke opdrachten en bevoegdheden toe.
- Bewerk sudoers met visudo; dit controleert de syntaxis vóór opslag.
- Controleer beleidswijzigingen zorgvuldig.

6. Veilige gebruikersbeheerprincipes

Beheer accounts en rechten bewust en regelmatig.

- Gebruik individuele accounts.
- Geef minimale noodzakelijke rechten en passende groepslidmaatschappen.
- Trek ongebruikte accounts in.
- Herzie periodiek machtigingen en sudo-toegang.

7. Veelvoorkomende risico's en controles

Te ruime of verouderde rechten vergroten de kans op ongewenste toegang.

- Controleer onverwachte schrijfrechten, speciale bits en brede sudo-toegang.
- Beperk toegang tot noodzakelijke gebruikers en opdrachten.
- Controleer wijzigingen en herzie rechten periodiek.

8. Korte samenvatting

Controleer steeds wie toegang heeft, welke handeling is toegestaan en waarom.

- Lees rechten per eigenaar, groep en overige gebruikers.
- Gebruik rwx en numerieke waarden correct.
- Beperk SUID, SGID en sudo tot wat noodzakelijk is; controleer de configuratie.

Linux-hardening en verplichte toegangscontrole

1. Wat is hardening?

- Verklein de blootstelling en mogelijke gevolgen van misbruik.
- Beperk onnodige functies, interfaces en kernelmodules.
- Hardening is een extra verdedigingslaag, geen garantie.

2. DAC versus MAC

- DAC: toegang op basis van eigenaarschap en machtigingen.
- Voorbeelden: gebruiker-, groep- en overige-rechten, ACL's en root-rechten.
- MAC: beleidssysteemregels begrenzen toegang naast of in plaats van DAC.
- DAC en MAC kunnen samen worden toegepast.

3. SELinux: contexten en labels

- Beveiligingscontexten koppelen kenmerken aan processen en objecten.
- Een context bevat onder meer een type.
- Labels beschrijven; het beleid bepaalt welke interacties zijn toegestaan.

4. SELinux: domeinen en beleid

- Processen kunnen in domeinen worden beperkt.
- Type-enforcementbeleid regelt toegang tot gelabelde objecten.
- Een proces kan worden begrensd, ook als DAC toegang toestaat.

5. AppArmor: profielen en modi

- Profielen beperken programma's op basis van regels, vaak gekoppeld aan paden.
- Handhavingsmodus kan overtredingen blokkeren.
- Complain-modus registreert overtredingen zonder ze te blokkeren.

6. Kernelhardening

- Beperk onnodige kernelinterfaces en modules.
- Beheer beveiligingsinstellingen zorgvuldig.
- Aanvullend op toegangscontrole; geen vervanging ervan.

7. Praktijkvoorbeeld en controle

- Een webserver krijgt via DAC alleen toegang tot benodigde bestanden.
- Een SELinux-domein of AppArmor-profiel kan toegang tot bestanden en andere systeembronnen verder beperken.
- Controleer welk beleid actief is en welke toegang het toestaat.

8. Korte samenvatting

- DAC gebruikt eigenaarschap en machtigingen; MAC voegt beleidsregels toe.
- SELinux werkt doorgaans met labels, domeinen en type-enforcementbeleid.
- AppArmor werkt doorgaans met profielregels, vaak op paden gebaseerd.
- Kernelhardening verkleint risico's en vult toegangscontrole aan.

Veilige softwareontwikkeling (SSDLC) & DevSecOps

1. SSDLC & DevSecOps

SSDLC = levenscyclus voor veilige softwareontwikkeling. DevSecOps integreert beveiliging in dagelijks ontwikkel- en beheerproces.

- Beveiliging bij ontwerp, bouw, test en oplevering.
- Ontwikkeling, beveiliging en beheer delen verantwoordelijkheid.

2. Security by Design

Neem beveiliging vanaf het ontwerp mee, niet pas achteraf.

- Beperk toegang tot wat nodig is.
- Kies veilige standaardinstellingen.
- Valideer invoer en bescherm gegevens.

3. SAST

SAST (statische applicatiebeveiligingstest) analyseert broncode of bytecode, doorgaans zonder uitvoering.

- Signaleert mogelijke codeproblemen vroeg.
- Beoordeel meldingen; niet elke melding is een kwetsbaarheid.
- Combineer met andere testvormen.

4. DAST

DAST (dynamische applicatiebeveiligingstest) onderzoekt een draaiende applicatie via interfaces.

- Let op waarneembaar risicogedrag.
- Test in een passende testomgeving.
- Vult codeanalyse aan, maar bewijst geen afwezigheid van kwetsbaarheden.

5. SCA & afhankelijkheden

SCA (analyse van softwarecomponenten) brengt afhankelijkheden in kaart en kan bekende kwetsbaarheden en licentie-informatar tonen.

- SBOM (softwareonderdelenlijst) inventariseert componenten.
- Bekijk bij een melding componentversie en gebruik.
- Controleer of een geschikte update beschikbaar is.
- Beoordeel herkomst en risico's van componenten.

6. CI/CD-controles

Een CI/CD-pijplijn automatiseert integratie, testen en levering of implementatie; beveiligingscontroles kunnen daarin.

- Voer bijvoorbeeld SAST en SCA uit bij wijzigingen.
- Gebruik DAST in een testomgeving.
- Stel meldingen en beoordelingsstappen in; volg bevindingen op.

7. Bevindingen opvolgen

Een testmelding is een signaal om te onderzoeken; beoordeel de context voordat je beslist.

- Leg vast wat is gevonden en waar.
- Beoordeel impact, gebruik en oplossing.
- Wijs een vervolgactie toe en controleer na aanpassing opnieuw.

Let op: Geen enkele afzonderlijke testmethode vindt noodzakelijkerwijs alle kwetsbaarheden.

8. Korte samenvatting

Bouw beveiliging in vanaf het ontwerp en blijf toetsen door de hele ontwikkelcyclus.

- Security by Design; voorkom risico's met veilige ontwerpkeuzes.
- SAST kijkt naar code; DAST test een draaiende applicatie.
- SCA maakt afhankelijkheden en componentrisico's zichtbaar.
- CI/CD kan controles automatiseren; mensen beoordelen en volgen meldingen op.

Softwaretoeleverings ketenbeveiliging en SBOM

1 Softwaretoeleveringsketenbeveiliging

Bescherm software van componenten tot bouwen, verspreiden en bijwerken.

- Risico's ontstaan in componenten, ontwikkel- en bouwprocessen, distributie en updates.

2 SBOM

Een Software Bill of Materials (SBOM) is een gestructureerde inventaris van softwarecomponenten en afhankelijkheden.

- Helpt producten met kwetsbare componenten op te sporen.
- Een inventaris is een hulpmiddel, geen veiligheidsgarantie.

3 Afhankelijkheden

Een afhankelijkheid is een component waarop software steunt.

- Direct: het project gebruikt de component zelf.
- Indirect: meegebracht door een andere afhankelijkheid.
- Inventariseer en werk waar nodig bij.

4 Dependency confusion

Bij dependency confusion raakt een pakketbeheerder mogelijk in verwarring tussen interne en publieke pakketnamen of versies.

- Een project kan onbedoeld een externe afhankelijkheid ophalen.
- Controleer pakketbronnen en hanteer eenduidige naam- en versieafspraken.

Voorbeeld

Een project verwacht een intern pakket, maar haalt door onduidelijke bron- of naaminstellingen een gelijknamig pakket van een niet-vertrouwde bron op.

5 Aanvallen op de bouwpijplijn

Een bouwpijplijn voert stappen uit om software te maken en klaar te zetten voor gebruik.

- Aanvallen richten zich op systemen, rechten, configuratie, geheimen of artefacten.
- Een gecompromitteerde stap kan de integriteit aantasten.
- Beperk toegang en bescherm geheimen.

6 Integriteitsverificatie

Controleer of een softwareartefact overeenkomt met een vertrouwde verwachte waarde of herkomst.

- Een cryptografische hash maakt wijzigingen aantoonbaar door vergelijking met een vertrouwde hashwaarde.
- Een digitale handtekening ondersteunt herkomst en integriteit als sleutel en verificatieketen betrouwbaar zijn.
- Hashes en handtekeningen zijn hulpmiddelen, geen bewijs dat software veilig is.

7 Verdediging in de praktijk

Pas controlemaatregelen toe op componenten en bouw- en distributieprocessen.

- Inventariseer en werk afhankelijkheden bij.
- Controleer gebruikte pakketbronnen.
- Beperk toegangsrechten en bescherm geheimen in bouwomgevingen.
- Verifieer softwareartefacten vóór gebruik.

8 Korte samenvatting

Beveilig de hele route van component tot software-update.

- Een SBOM maakt componenten en afhankelijkheden inzichtelijk.
- Let op verwarring rond pakketnamen en aanvallen op bouwsystemen.
- Controleer integriteit en herkomst, maar beschouw controles niet als veiligheidswaarborg.

Ethisch hacken: fasen en methodologie

Didactisch model voor een geautoriseerde penetratietest. Fasen en benamingen kunnen verschillen.

Altijd binnen de afspraken:
vooraf schriftelijke toestemming • afgebakende scope • afgesproken tijdvenster • beperkte impact • volg stop- en escalatieafspraken.

1. Wat is het

- Afgesproken beoordeling van beveiliging.
- Didactisch model, niet universeel.

2. Doel en toestemming

- Schriftelijke toestemming.
- Leg scope, tijdvenster en stop-/escalatieafspraken vast.
- Beperk impact.

3. Verkenning

- Verzamel informatie binnen scope.
- Passief: bestaande info.
- Actief: benadert systemen, passende toestemming vereist.

4. Scannen

- Onderzoek alleen geautoriseerde systemen.
- Vind mogelijke zwakke plekken.
- Bevinding ≠ bewijs van exploiteerbaarheid.

5. Toegang verkrijgen

- Valideer gevolgen alleen binnen expliciete toestemming en grenzen.
- Geen exploitatiestappen.

6. Toegang behouden

- Alleen gecontroleerd beoordelen.
- Vermijd wijzigingen en persistentie op productie.

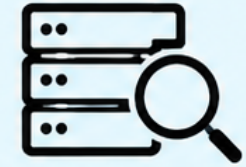
7. Sporen wissen en rapportage

- **BELANGRIJK:** deze historische fasebenaming geeft géén toestemming om logs of bewijs te verwijderen.
- Ruim alleen afgesproken testartefacten op; behoud logs.
- Rapporteer scope, aanpak, bevindingen, impact en hersteladvies.

8. Korte samenvatting

- Toestemming en scope.
- Verken, scan, valideer zorgvuldig.
- Stop bij onverwachte impact of buiten-scope toegang.
- Rapporteer transparant.

Poortscanning & Netwerkkartering (Nmap Grondbeginselen)



1. Wat is het

Poortscanning onderzoekt welke netwerkpoorten vanaf een bepaald meetpunt bereikbaar lijken. Nmap is een hulpmiddel voor netwerkdetectie en beveiligingsinventarisatie.

- Een poort is een logisch communicatie-eindpunt voor netwerkverkeer.
- Een scan toont wat de scanner kan waarnemen; routering, filtering en systeeminstellingen beïnvloeden de uitkomst.

2. Waarom inventariseren

Een geautoriseerde inventarisatie helpt beheerders zicht te krijgen op bereikbare systemen en netwerkdiensten.

- Vergelijk waargenomen diensten met de goedgekeurde inventaris.
- Onderzoek onverwachte resultaten via een veilige, toegestane werkwijze.
- Voer scans alleen uit op eigen systemen of met uitdrukkelijke toestemming.

3. TCP SYN-scan

De scan stuurt een eerste TCP-SYN-pakket en beoordeelt de reactie zonder de gewone TCP-handshake af te maken. Daarom heet dit ook een halfopen scan.

- **SYN/ACK:** aanwijzing dat de poort open is.
- **Reset (RST):** aanwijzing dat de poort gesloten is.
- **Geen reactie of een filtermelding:** gefilterd of onduidelijk; dit bewijst niet dat de poort open is.

4. TCP Connect-scan

Een connect-scan vraagt het besturingssysteem om volgens de normale TCP-procedure verbinding te maken.

- Een geslaagde verbinding wijst erop dat de poort bereikbaar en open is.
- Een afgewezen verbinding kan erop wijzen dat de poort gesloten is.
- Bij een geslaagde verbinding wordt de gewone handshake voltooid; dat onderscheidt deze scan van de SYN-scan.

5. UDP-scan

UDP heeft geen TCP-achtige handshake. Daardoor kan een open UDP-poort stil blijven wanneer een scan deze onderzoekt.

- Een ICMP-melding 'poort onbereikbaar' wijst erop dat de poort gesloten is.
- Stilte kan de uitkomst 'open|gefilterd' opleveren: de scan kan niet vaststellen welke van de twee geldt.
- Stilte bewijst niet dat er een dienst actief is.

6. Service-detectie

Service- en versiedetectie vergelijkt reacties op onderzoekspakketten om de dienst en, waar mogelijk, de versie af te leiden.

- Gebruik de uitkomst als aanwijzing voor een inventarisatie.
- Controleer belangrijke identificaties afzonderlijk; detectie is geen garantie van exacte herkenning.

7. Resultaten interpreteren

Een scanresultaat beschrijft wat vanaf het meetpunt zichtbaar was, niet noodzakelijk de volledige toestand van het doelnetwerk.

- Houd rekening met routering, filters en configuratie van het doelsysteem.
- Maak onderscheid tussen 'open', 'gesloten', 'gefilterd' en een onduidelijke uitkomst.
- Bevestig afwijkingen met toegestane inventarisatie-gegevens en context.

8. Korte samenvatting

Poortscans geven beheerders een momentopname van bereikbare netwerkpoorten en mogelijke diensten.

- **SYN-scan:** beoordeelt reacties zonder de gewone handshake af te maken.
- **Connect-scan:** gebruikt de normale TCP-verbindingsprocedure.
- **UDP-scan:** stilte laat de poortstatus vaak onbeslist.
- **Scan alleen systemen waarvoor je bevoegdheid hebt.**

Veilig gebruik

Oefen in een gecontroleerd lab of scan uitsluitend systemen die je bezit of waarvoor je uitdrukkelijke toestemming hebt.

Webapplicatie-penetratietesten & Burp Suite

1. Wat is het

- Een onderscheppende proxy staat tussen testclient en webapplicatie.
- Bekijk verzoeken en antwoorden; pas ze alleen aan binnen een geautoriseerde test.

2. Waarom testen?

- Controleer of invoer en toegang correct worden gevalideerd.
- Spoor onbedoelde blootstelling of zwakke sessieafhandeling op.

3. Proxy-tools

- Burp Suite en OWASP ZAP kunnen HTTP(S)-verkeer inspecteren.
- Ze ondersteunen handmatige webbeveiligingstests; ze garanderen geen volledige beveiliging.

4. Testopstelling

- Gebruik een testclient, proxy en webapplicatie waarvoor je toestemming hebt.
- Laat testverkeer via de proxy lopen en bekijk verzoeken en antwoorden.

Alleen met toestemming
Test uitsluitend systemen, accounts en gegevens waarvoor je expliciete toestemming hebt.

5. Invoervalidatie

- Vergelijk gecontroleerde varianten van dezelfde invoer.
- Controleer of de server invoer valideert; validatie in de browser alleen is niet genoeg.
- Voorbeeld: vergelijk hoe een formulier reageert op een geldige en een ongeldige waarde.

6. Sessiebeheer

- Bekijk sessiecookies en relevante verzoeken.
- Controleer het gedrag bij aanmelden, uitloggen en sessievernieuwing.

7. API-endpoints

- Inspecteer verzoeken en antwoorden van toegestane API-endpoints.
- Let op parameters, authenticatie en toegangscontrole.
- Gebruik uitsluitend goedgekeurde accounts en testgegevens.

8. Korte samenvatting

- Een proxy maakt webverkeer zichtbaar voor handmatige inspectie.
- Onderzoek invoervalidatie, sessiebeheer en API-toegang zorgvuldig.
- Werk alleen binnen de afgesproken testscope.

Privilegebescherming en lokale escalatievectoren

1. Wat is privilege-escalatie?

Privilege-escalatie is het verkrijgen van meer rechten dan een gebruiker of proces hoort te hebben. Lokaal gebeurt dit binnen één computer of besturingssysteem.

- **Verticaal:** een hoger toegangsniveau verkrijgen, bijvoorbeeld van standaardgebruiker naar beheerder of root.
- **Horizontaal:** toegang krijgen tot een andere account met een vergelijkbaar toegangsniveau.

2. Windows-context

Windows-beheerdersrechten geven ruime controle over het systeem. Gebruik dagelijkse accounts waar mogelijk met beperkte rechten.

- Controleer wie lid is van beheerdersgroepen.
- Beoordeel bestands- en mapmachtigingen en serviceconfiguraties.
- Houd updates bij en bekijk beveiligingslogs.

3. Linux-context

Rootrechten geven ruime controle over een Linux-systeem. Voor dagelijks werk zijn accounts met beperkte rechten doorgaans passender.

- Controleer gebruikers, groepen en bestandsrechten.
- Beoordeel of services alleen de rechten hebben die ze nodig hebben.
- Houd updates bij en bekijk systeemlogs.

8. Korte samenvatting

Lokale privilege-escalatie betekent dat een gebruiker of proces meer rechten krijgt dan bedoeld. Beperk rechten, controleer accounts, bestanden en services, werk software bij en volg wijzigingen op.

- **Verticaal** betekent een hoger toegangsniveau; horizontaal betekent een andere account op vergelijkbaar niveau.
- Windows-beheerdersrechten en Linux-rootrechten vragen om zorgvuldige beperking.
- Gebruik veilige, niet-operationele voorbeelden en werk alleen op geautoriseerde systemen.

4. Lokale risicocategorieën

Onveilige instellingen kunnen onbedoeld te ruime toegang mogelijk maken.

- Bestanden of mappen met te ruime machtigingen.
- Overmatige rechten voor accounts of groepen.
- Onveilig ingerichte services.
- Verouderde software.
- Onbeschermde inloggegevens.

7. Controle- en herstelcyclus

Voer controles uitsluitend uit op systemen **waarvoor** je bevoegd bent. Schermnamen en **beheeropties** verschillen per Windows-versie en Linux-distributie.

1. **Inventariseer** — Breng accounts, groepen en services in kaart.
2. **Controleer** — Beoordeel rechten, configuraties en updates op onnodige privileges of onveilige instellingen.
3. **Herstel** — Pas onnodige rechten en onveilige instellingen aan en installeer passende updates.
4. **Verifieer** — Controleer de wijziging met logs en plan periodieke hercontrole.

6. Verdedigingsmaatregelen

Beperk de kans op rechtenmisbruik door onnodige toegang te verminderen en wijzigingen te controleren.

- Pas minimale rechten toe; geef alleen toegang die nodig is voor de taak.
- Scheid standaard- en beheerdersaccounts.
- Gebruik sterke authenticatie en bescherm inloggegevens.
- Beheer patches tijdig en gebruik veilige machtigingen.
- Beperk serviceprivileges en controleer wijzigingen en logs.

Belangrijk
Geen enkele maatregel biedt op zichzelf een volledige beveiligingsgarantie. Stem controles af op het systeem en de omgeving.

5. Signalen en voorbeelden

Herken configuratierisico's zonder ze als exploitstappen te behandelen.

- Een standaardgebruiker kan een bestand wijzigen dat door een bevoorrecht proces wordt gebruikt.
- Een service draait met meer rechten dan voor haar functie nodig is.
- Dit zijn aanwijzingen om instellingen te laten beoordelen en herstellen; ze bewijzen op zichzelf geen misbruik.

SOC — architectuur en rollen

1. Wat is een SOC?

Een **Security Operations Center** (SOC) is een organisatorische functie die beveiligingssignalen volgt en meldingen beoordeelt. Het helpt mogelijke incidenten te detecteren, onderzoeken en af te handelen.

- **Mensen**, processen en beveiligingstechnologie werken samen.
- De inrichting verschilt per organisatie.

2. Doelen en taken

Een SOC helpt risico's zichtbaar te maken en passende opvolging te organiseren.

- **Monitoren:** signalen uit systemen en beveiligingstools volgen.
- **Beoordelen:** context verzamelen en meldingen prioriteren.
- **Escaleren:** werk doorgeven aan de juiste rol of het juiste team.
- **Leren:** incidenten gebruiken om detecties en processen te verbeteren.

3. Organisatie en teams

Een SOC kan uit verschillende rollen bestaan. Een veelgebruikt model verdeelt het werk over Tier 1, Tier 2 en Tier 3, met ondersteuning van een manager en andere teams.

- De precieze taakvercdeling en bezetting verschillen per organisatie.
- Afstemming met IT, netwerkbeheer en leidinggevenden kan nodig zijn.

4. Tier 1 — monitoring en triage

Tier 1 is vaak het eerste aanspreekpunt voor beveiligingsmeldingen. Triage betekent een melding beoordelen en bepalen welke opvolging passend is.

- Controleert waarschuwingen en verzamelt basiscontext.
- Legt bevindingen en uitgevoerde acties vast.
- Escalatie is passend als er aanwijzingen zijn voor een echt incident of meer onderzoek nodig is.

5. Tier 2 — onderzoek en respons

Tier 2 onderzoekt meldingen die meer diepgang vragen en helpt bij de incidentrespons.

- Combineert signalen en context om gebeurtenissen beter te begrijpen.
- Onderzoekt mogelijke impact en coördineert passende vervolgstappen.
- Werkt samen met Tier 1 en relevante technische teams.

6. Tier 3 — specialistische analyse

Tier 3 kan worden ingeschakeld bij complexe dreigingen of wanneer specialistische analyse nodig is.

- Kan diepgaand onderzoek en threat hunting uitvoeren.
- Helpt complexe of organisatiebrede incidenten te begrijpen.
- De precieze rol en inzet verschillen per organisatie.

7. SOC-manager en ploegendiensten

De **SOC-manager** coördineert mensen, processen en prioriteiten en stemt af met andere teams en leidinggevenden. De manager hoeft niet elk technisch onderzoek zelf uit te voeren.

- Een ploegendienstmodel verdeelt monitoring over toegewezen diensten.
- Bij een overdracht worden openstaande meldingen, incidentstatus, acties en verantwoordelijkheden duidelijk vastgelegd.
- Een dienstrooster of bezetting is afhankelijk van de organisatie.

8. Korte samenvatting

Een SOC volgt beveiligingssignalen, beoordeelt meldingen en helpt incidenten af te handelen. Tier 1 doet vaak eerste triage, Tier 2 verdiept het onderzoek en Tier 3 kan specialistische analyse leveren. De SOC-manager coördineert het werk; duidelijke overdrachten ondersteunen samenwerking.

- **Voorbeeld**: Tier 1 beoordeelt een endpointwaarschuwing en verzamelt basiscontext.
- Bij aanwijzingen voor een incident escaleert Tier 1 naar Tier 2 voor correlatie en onderzoek.
- Bij een complexe of organisatiebrede dreiging kan Tier 3 specialistische analyse uitvoeren.

SIEM-systemen: logverzameling, correlatie en detectie

1. Wat is SIEM?

- Beveiligingsgebeurtenissen uit meerdere bronnen samenbrengen
- Ondersteunt onderzoek en detectie
- Bruikbaarheid hangt mede af van kwaliteit en volledigheid van logs

2. Logbronnen

- Besturingssystemen
- Netwerkapparatuur
- Toepassingen en identiteitsdiensten
- Beschikbare gegevens verschillen per bron

3. Verzameling

- Logaanvoer brengt gebeurtenissen naar SIEM
- Eerste stap in de workflow
- Ontbrekende logs kunnen beoordeling beperken

4. Parsing

- Ruwe records opdelen in verwerkbare gegevens
- Bijvoorbeeld: tijdstip, bron, gebruiker, actie, resultaat
- Velden verschillen per bron

5. Normalisatie

- Gegevens onderbrengen in een consistente veldindeling
- Helpt vergelijkbare gebeurtenissen samen te onderzoeken
- Afhankelijk van beschikbare bronvelden

6. Correlatieregels

- Gebeurtenissen combineren volgens criteria
- Tijdsvenster, overeenkomstige velden of volgorde
- Voorbeeld: meerdere mislukte aanmeldingen, daarna een geslaagde aanmelding voor hetzelfde account

Belangrijk
Signaal voor nader onderzoek, geen bewijs op zichzelf van een inbraak

7. Detectie en triage

- Waarschuwingen op basis van gebeurtenissen en correlaties
- Beoordeel gebeurtenissen en context
- Een waarschuwing alleen bewijst geen inbraak

8. Korte samenvatting

- Verzamelen → parsen en normaliseren → detecteren → beoordelen
- Logkwaliteit en volledigheid beïnvloeden bruikbaarheid
- Correlatie helpt signalen samen te bekijken

068. SOAR: beveiligingsorkestratie, automatisering en respons

1. Wat is SOAR?

SOAR staat voor Security Orchestration, Automation and Response: beveiligingsorkestratie, automatisering en respons. Een SOAR-platform kan beveiligingstools en workflows verbinden, routinetaken automatiseren en incidentrespons ondersteunen.

- **Orkestratie:** systemen en stappen op elkaar afstemmen.
- **Automatisering:** herhaalbare taken volgens ingestelde regels uitvoeren.
- **Respons:** informatie en goedgekeurde acties rond een beveiligingsincident coördineren.

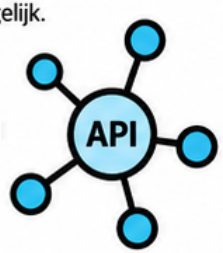
2. Bouwstenen (SIEM, EDR, firewall)

SOAR kan informatie uit verschillende beveiligingssystemen samenbrengen. Uitwisseling en aansturing hangen af van beschikbare integraties en instellingen.

- **SIEM:** verzamelt en analyseert beveiligingsgebeurtenissen.
- **EDR:** bewaakt en helpt activiteit op eindpunten, zoals computers, onderzoeken.
- **Firewall:** regelt netwerkverkeer aan de hand van ingestelde regels.

3. Orkestratie en API-koppelingen

Orkestratie verbindt informatie en werkstappen van meerdere systemen tot een samenhangende workflow. API-koppelingen maken programmatische verzoeken en gegevensuitwisseling mogelijk.



- Een koppeling vereist passende authenticatie en machtigingen.
- Beschikbare acties verschillen per API en productconfiguratie.
- Niet elke actie is automatisch beschikbaar of toegestaan.

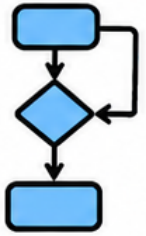
4. Playbook-opbouw

Een playbook beschrijft een herhaalbare workflow. Duidelijke voorwaarden, uitzonderingen en registratie maken stappen begrijpelijk en controleerbaar.

1. **Waarschuwing ontvangen** — start bij een melding.
2. **Context verzamelen** — haal relevante informatie op uit gekoppelde systemen.
3. **Criteria controleren** — toets vooraf bepaalde voorwaarden.
4. **Actie voorleggen of uitvoeren** — alleen met passende configuratie en autorisatie.
5. **Uitzonderingen afhandelen** — bepaal wat te doen bij ontbrekende informatie, onjuiste criteria of vastlopen.
6. **Resultaat vastleggen** — registreer uitkomst en relevante stappen.

5. Voorbeeld van een responsworkflow

Conceptueel voorbeeld: een SIEM-melding kan via een playbook worden verrijkt met beschikbare EDR-context. Na beoordeling kan een analist een beperkte, geautoriseerde responsactie laten uitvoeren en het resultaat laten registreren.



- SIEM-melding ontvangen → relevante EDR-context verzamelen.
- Criteria controleren en bevindingen beoordelen.
- Een toegestane actie ter goedkeuring voorleggen of volgens de configuratie uitvoeren.
- Resultaat registreren; precieze stappen hangen af van systemen, instellingen en bevoegdheden.

6. Menselijke controle, rechten en logging

Automatisering vervangt menselijke beoordeling niet vanzelf. Passende rechten, goedkeuringsmomenten en registratie houden acties binnen afgesproken bevoegdheden en maken beoordeling achteraf mogelijk.

- Geef koppelingen alleen passende machtigingen.
- Bepaal welke acties voorafgaande goedkeuring vereisen.
- Leg relevante beslissingen en resultaten vast.
- Controleer uitzonderingen en afwijkende acties.

Let op: een actie is alleen mogelijk en toegestaan als API, configuratie en autorisatie dat ondersteunen.

7. Voordelen en aandachtspunten

SOAR kan herhaalbare werkstappen stroomlijnen en informatie samenbrengen. De waarde hangt af van zorgvuldige inrichting en passende controle.

- **Voordeel:** minder handmatig werk bij routinetaken.
- **Voordeel:** informatie en stappen komen samen in één workflow.
- **Aandachtspunt:** onjuiste voorwaarden kunnen tot ongeschikte vervolgstappen leiden.
- Integraties, acties en rechten verschillen per product en configuratie.
- Houd uitzonderingen, goedkeuring en registratie in beeld.

8. Korte samenvatting

SOAR verbindt beveiligingstools en workflows om respons te ondersteunen en sommige routinetaken te automatiseren.

- **SIEM** analyseert gebeurtenissen; EDR bewaakt eindpunten; een firewall regelt netwerkverkeer.
- **API's** ondersteunen uitwisseling binnen grenzen van authenticatie, rechten en configuratie.
- Een goed playbook bevat voorwaarden, uitzonderingen en registratie.
- Automatische acties vereisen autorisatie en passende menselijke controle.

Incidentresponslevenscyyclus (NIST SP 800-61)

De acht onderdelen in deze mindmap zijn een praktische uitsplitsing van de vier officiële NIST-fasen, geen acht afzonderlijke fasen.

1. Wat is incidentrespons?

- Samenhangende aanpak: voorbereiden, herkennen, beperken en leren.
- Vier NIST-fasen.
- Acht onderdelen zijn praktische uitsplitsing, geen officiële fasen.

Voorbeeld: Melding over mogelijk besmette werkplek: onderzoek en reageer stap voor stap.

2. Voorbereiding

- Stel rollen, verantwoordelijkheden en contactpersonen vast.
- Richt communicatie, escalatie en logging in.
- Oefen met scenario's en zorg dat procedures bekend zijn.

3. Detectie

- Herken meldingen en alerts uit verschillende bronnen.
- Een alert is een aanwijzing, geen bewijs.
- Verzamel tijdstippen, apparaten en logbestanden.
- Leg vast wie de eerste melder was.

4. Analyse

- Valideer meldingen en zoek naar gerelateerde signalen.
- Bepaal welke systemen, gegevens en gebruikers zijn getroffen.
- Documenteer bevindingen en onzekerheden.

NIST definieert vier fasen. Deze mindmap geeft dieper inzicht met acht onderdelen.

Zelfde doel, sterkere organisaties: van melding naar herstel en blijvend leren.

5. Inperking

- Beperk de impact van het incident.
- Kies maatregelen die passen bij de situatie.
- Tijdelijke netwerkisolatie kan verspreiding beperken.
- Leg acties en overwegingen vast.
- **Let op:** er is geen universele procedure; maatwerk is nodig.

6. Uitroeiing en herstel

- Verwijder kwaadaardige sporen en pak de oorzaak aan.
- Herstel systemen vanuit vertrouwde bronnen.
- Controleer op herhaling van het incident.

7. Evaluatie na het incident

- Leg de tijdlijn, beslissingen, acties en uitkomsten vast.
- Bespreek wat goed ging en waar vertraging ontstond.
- Verbeter procedures en oefen met nieuwe inzichten.

8. Korte samenvatting

- **NIST kent vier fasen:**
  1. Voorbereiding
  2. Detectie en analyse
  3. Inperking, uitroeiing en herstel
  4. Evaluatie na het incident
- Deze mindmap splitst de fasen uit in acht praktische onderdelen.
- Leerervaringen leiden tot verbeteringen in mensen, processen en technologie.

De vier fasen van NIST SP 800-61

Digitale forensica (DFIR) & keten van bewaring

1. Wat is DFIR?

- DFIR: digitaal forensisch onderzoek en incidentrespons.
- Onderzoekt digitale gegevens zorgvuldig en reproduceerbaar.
- Helpt gebeurtenissen reconstrueren en bevindingen onderbouwen.

2. Doel en principes

- Beperk wijzigingen aan de oorspronkelijke gegevensbron.
- Leg noodzakelijke handelingen en waarnemingen vast.
- Bewaar het origineel veilig; onderzoek waar mogelijk een gecontroleerde kopie.
- Een schrijfblokkering kan wijzigingen aan opslagmedia helpen voorkomen.

3. Volgorde van vluchtigheid

- Prioriteer gegevens die snel kunnen verdwijnen.
- Voorbeelden: actieve netwerkverbindingen en vluchtig geheugen.
- Verzamel daarna duurzamere gegevens, zoals bestanden op opslagmedia.
- De praktische volgorde hangt af van het systeem en de situatie.

4. Vastleggen en documenteren

- Noteer de context en relevante waarnemingen.
- Registreer welke handelingen zijn uitgevoerd en wanneer.
- Leg vast wie handelde en waar gegevens werden bewaard.
- Documentatie maakt het verloop van het onderzoek navolgbaar.

5. Forensische kopieën

- Een forensische kopie is een bit-voor-bit-kopie van een gegevensbron.
- Gebruik een gecontroleerde kopie als werkexemplaar voor analyse.
- Behoud het origineel veilig en beperk wijzigingen aan de bron.

6. Integriteit met hashes

- Een cryptografische hash, zoals SHA-256, vat gegevens samen in een vaste reeks tekens.
- Vergelijk hashes van de bron en de kopie om overeenkomst te controleren.
- Een gelijke hash ondersteunt de integriteitscontrole.
- Een hash bewijst op zichzelf niet hoe of door wie gegevens zijn verkregen.

7. Keten van bewaring

- De keten van bewaring (chain of custody) documenteert de geschiedenis van digitaal bewijsmateriaal.
- Registreer verzamelen, overdragen, opslaan, openen en onderzoeken.
- Noteer per handeling wie deze uitvoerde, wanneer, wat er gebeurde en waar het materiaal werd bewaard.

8. Korte samenvatting

- Voorbeeld: een organisatie onderzoekt een melding van verdachte activiteit.
- Leg eerst de context vast; verzamel relevante vluchtige gegevens als dat nodig is.
- Maak en controleer een forensische kopie, registreer overdrachten en analyseer de kopie.
- Zo blijven het onderzoeksverloop en de omgang met het origineel navolgbaar.

Cyberdreigingsinformatie (CTI) & dreigingsfeeds

1 Wat is CTI?

- Informatie over dreigingen
- Helpt inzicht in risico's en beveiligingsbeslissingen
- Actoren, campagnes, technieken en indicatoren

2 Drie niveaus van CTI

- Strategisch: trends, actoren, gevolgen
- Operationeel: campagnes, doelen, werkwijzen
- Tactisch: technieken en waarneembare indicatoren

3 Dreigingsfeeds

- Periodieke of doorlopende informatie
- Voorbeelden: domein, IP-adres, bestands-hash
- Beoordeel bron, context, ouderdom en betrouwbaarheid

4 IoC's

- Indicator van mogelijk incident
- Voorbeelden: bestands-hash, domeinnaam, IP-adres, e-mailkenmerk
- Contextafhankelijk, geen sluitend bewijs op zichzelf.

Belangrijk
Trek geen conclusie op basis van één indicator zonder context en aanvullende beoordeling.

5 STIX en TAXII

- STIX beschrijft en structureert informatie
- TAXII ondersteunt uitwisseling
- Complementair maar verschillend

6 Verzamelen en beoordelen

- Controleer bron, context, actualiteit en betrouwbaarheid
- Controleer wanneer de indicator voor het laatst is waargenomen
- Weeg relevantie voor eigen systemen
- Beoordeel vóór detectieregel of actie

7 Gebruik in verdediging

- Input voor monitoring en detectie
- Koppel indicatoren aan context
- Werk detecties bij bij veroudering

8 Korte samenvatting

- Drie CTI-niveaus
- Indicatoren zijn aanwijzingen, geen bewijs
- STIX structureert, TAXII wisselt uit
- Beoordelen vóór toepassen

Proactief dreigingsjagen

1 Wat is dreigingsjagen?

- Onderzoek naar mogelijke kwaadwillige activiteit.
- Zoekt signalen die controles missen.
- Vult monitoring en respons aan.

2 Hypothese vormen

- Begin met een toetsbare verwachting.
- Bepaal bereik en ondersteunende of ontkrachtende gegevens.

3 Anomalieën herkennen

- Afwijking is geen bewijs van inbraak.
- Vergelijk met een passende basislijn.
- Onderzoek onschuldige verklaringen.

4 Gegevensbronnen

- Apparaat-, proces-, account-, DNS- en netwerkgegevens.
- Noteer bronnen en periode.
- Ontbrekende gegevens beperken conclusies.

5 Zoeken en toetsen

- Verbind signalen tussen bronnen, apparaten, accounts en tijden.
- Toets waarnemingen aan de hypothese.
- Context bepaalt vervolgonderzoek.

6 Bevindingen beoordelen

- Beoordeel signalen in context.
- Controleer normale of onschuldige oorzaken.
- Noteer onzekerheid en ontbrekende gegevens.

7 Resultaat en opvolging

- Uitkomst: onschuldig, onvoldoende gegevens, betere detectie of opvolging.
- Bewaar waarnemingen; communiceer onzekerheid.
- Volg bestaand incidentresponsproces.

8 Korte samenvatting

- Formuleer hypothese of start met afwijking.
- Kies gegevens; verbind gebeurtenissen; onderzoek context.
- Leg bronnen, periode, bevindingen en onzekerheid vast.

Containerbeveiliging & Docker-hardening

1. Wat is containerbeveiliging?

Containers isoleren processen, maar delen de kernel van het hostbesturingssysteem. Beveiliging vraagt daarom om meerdere aanvullende maatregelen.

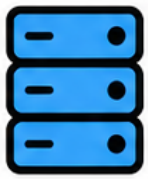
- Geen aparte kernel zoals bij een virtuele machine.
- Beperk wat een container kan zien en doen.
- Combineer veilige images, toegangsbeperking en runtime-instellingen.

2. Containerarchitectuur.

Een containerimage levert de bestanden en instellingen waaruit een container wordt gestart. De container draait als geïsoleerd proces op de host.

- **Image:** opgebouwd uit lagen en een basisimage.
- **Container:** draaiende instantie van een image.
- **Host:** levert de gedeelde kernel.

3. Linux-namespaces.



Namespaces geven processen verschillende weergaven van geselecteerde systeembronnen.

- **Netwerknamespace:** eigen netwerkinterfaces en netwerkweergave.
- **PID-namespace:** afzonderlijke weergave van proces-ID's.
- Mount- en hostname-namespaces beperken of scheiden andere weergaven.

4. Cgroups en resources.

Cgroups groeperen processen om hun resourcegebruik te registreren en waar nodig te begrenzen. Ze vullen namespaces aan, maar bieden een ander soort controle.

- Stel passende CPU- en geheugenlimieten in.
- Voorbeeld: geef een container een geheugenlimiet.
- Controleer gebruik en pas limieten aan de toepassing aan.

5. Basisimages en kwetsbaarheden.

Een basisimage kan verouderde software of onnodige pakketten bevatten. Dat kan kwetsbaarheden toevoegen en het aanvalsoppervlak vergroten.

- Kies een onderhouden, zo klein mogelijke basisimage.
- Neem alleen benodigde pakketten op.
- Bouw images opnieuw wanneer updates beschikbaar zijn.

6. Rootless Docker.

Rootless Docker voert de daemon en containers uit zonder host-rootrechten. Dit kan de gevolgen van bepaalde compromitteringen beperken, maar neemt risico's niet weg.

- Gebruik rootless waar dit past bij de omgeving.
- Beperk daarnaast rechten en blootgestelde diensten.
- Rootless vervangt geen andere beveiligingsmaatregelen.

7. Runtime-isolatie en hardening.

Rontimebeveiliging bestast uit maatregelen die gelden terwijl containers draaien. Ze verminderen risico's, maar garanderen geen volledige isolatie.

- Vermijd privileged-modus wanneer die niet nodig is.
- Beperk Linux-capabilities tot wat de toepassing nodig heeft.
- Gebruik passende beveiligingsprofielen en resource-limieten.

8. Korte samenvatting.

Containerbeveiliging berust op meerdere lagen die elkaar aanvullen.

- Containers delen de hostkernel.
- Namespaces scheiden geselecteerde systeemweergaven; cgroups beheren resourcegebruik.
- Gebruik onderhouden minimale images, rootless waar passend en runtime-hardening.

Kubernetes Beveiliging & Clusterbescherming

Een beveiligd cluster combineert toegangsbeheer, netwerkregels, veilige Pod-instellingen en controle van wijzigingen. Geen enkele maatregel vervangt de andere.

1. Wat is het

- Clusterbeveiliging beschermt de Kubernetes-omgeving en workloads.
- Beperk wie het cluster beheert.
- Beperk netwerkverkeer van en naar Pods.
- Pas veilige Pod-instellingen toe.
- Controleer wijzigingen vóór opslag.

2. API Server

- Centraal toegangspunt voor Kubernetes-API-verzoeken.
- Sta alleen bevoegde gebruikers en systemen toe.
- Authenticatie stelt identiteit vast.
- Autorisatie bepaalt toegestane acties.

3. RBAC

- Role-Based Access Control kent machtigingen toe via rollen en roltoewijzingen.
- Role geldt binnen één namespace.
- **ClusterRole** beschrijft clusterbrede of namespace-overstijgende machtigingen.
- Ken alleen noodzakelijke rechten toe.

4. Network Policies

- Beperken verkeer naar en van Pods met Pod-selectors en regels.
- Regels voor inkomend en uitgaand verkeer.
- Controleer ondersteuning door de netwerkplug-in.
- Regelen verkeer; vervangen geen authenticatie of encryptie.

5. Pod Security Standards

- Drie profielen met oplopende beperkingen voor Pod-instellingen.
- **Privileged**: minst beperkend.
- **Baseline**: basisbeperkingen.
- **Restricted**: meest beperkend.
- Beleid; afdwinging kan via admission control.

6. Admission Controllers

- Onderscheppen API-verzoeken vóór opslag van wijzigingen.
- Kunnen verzoeken valideren, weigeren of aanpassen.
- Beleidscontroles houden ongewenste configuraties tegen.
- Vullen toegangsbeheer en Pod-beveiligingsbeleid aan.

7. Praktische beveiligingskeuzes

- Beperk API-toegang tot bevoegde identiteiten.
- Geef via RBAC alleen noodzakelijke machtigingen.
- Beperk Pod-verkeer met NetworkPolicies waar ondersteund.
- Kies passend Pod Security Standards-profiel; richt zo nodig afdwinging in.
- Controleer configuraties met admission controllers vóór opslag.

Onthoud

Rechten, netwerkverkeer, Pod-instellingen en wijzigingscontroles zijn verschillende, aanvullende beveiligingslagen.

8. Korte samenvatting

- Beveilig de API-server; beperk machtigingen met RBAC; regel Pod-verkeer met NetworkPolicies; pas passende Pod-beveiligings-profielen toe; controleer wijzigingen met admission controllers.
- Elke maatregel heeft een eigen rol; samen bieden ze bredere bescherming.

Cloudbeveiliging: CSPM & CWPP

1. Wat is het?

- CSPM beoordeelt cloudconfiguraties op risico's en afwijkingen van gekozen richtlijnen.
- CWPP beveiligt workloads zoals virtuele machines, containers en serverless toepassingen.
- Samen: configuratie en beveiligings-houding tegenover workloads en gedrag.

Onthoud: Een CSPM-bevinding signaleert configuratie; het bewijst geen compromittering.

2. Waarom multicloud?

- Diensten kunnen uit meerdere cloudomgevingen komen.
- Centraliseer bevindingen om te bekijken, prioriteren en op te volgen.
- Zichtbaarheid hangt af van accounts, workloads, rechten en databronnen.
- Een dashboard garandeert geen volledige dekking.

3. CSPM

- Inventariseert cloudresources en beoordeelt configuraties.
- Signaleert afwijkingen van gekozen beveiligingsrichtlijnen.
- Voorbeelden: publieke opslag, te ruime rechten of logging uit.
- Monitoring herstelt niet automatisch veilig.

4. CWPP

- Beschermt virtuele machines, containers en serverless workloads.
- Mogelijk: kwetsbaarheidsbeheer, detectie van verdacht gedrag en runtimebescherming.
- Signalen: ontbrekende updates of onverwacht procesgedrag.
- Functies verschillen per product.

5. Zichtbaarheid en signalen

- Controleer aangesloten accounts, workloads en databronnen.
- Volg dekking, bevindingen per ernst, ouderdom en terugkerende misconfiguraties.
- Leg meetmethode en scope vast voor vergelijking.
- Meldingen vragen beoordeling.

6. Bevindingen prioriteren

- Beoordeel ernst, context, geraakt onderdeel en mogelijke impact.
- Let op blootstelling, gegevensgevoeligheid en beveiligingsmaatregelen.
- Scheid configuratiebevindingen van signalen over workloadgedrag.
- Wijs een eigenaar toe.

7. Opvolgen en naleving

- Leg controle-eis en resources binnen scope vast.
- Bewaar bewijs en beoordeel of dit de eis ondersteunt.
- Herstel volgens wijzigings- en toegangsprocedures; controleer opnieuw.
- Een toolmelding bewijst geen volledige naleving.

Veilige opvolging: Beoordeel, wijs een eigenaar toe en volg goedgekeurde procedures.

8. Korte samenvatting

- CSPM kijkt naar configuraties en beveiligingshouding; CWPP naar workloads.
- Gebruik beide als aanvulling, geen garantie op veiligheid of dekking.
- Inventariseer, beoordeel, prioriteer, wijs eigenaar toe, herstel zorgvuldig en controleer opnieuw.
- Mensen interpreteren signalen en valideren bevindingen.

Privacywetgeving: AVG/GDPR & LGPD

1. Wat is het

- Wetten voor verwerking van persoonsgegevens
- AVG: Europese Unie; GDPR is de Engelse afkorting
- LGPD: Braziliaans kader; bepalingen verschillen van de AVG

2. Persoonsgegevens

- Gegevens over een geïdentificeerde of identificeerbare persoon
- Voorbeelden: naam, e-mailadres of klantnummer
- Verwerking: verzamelen, opslaan, raadplegen, wijzigen, delen of verwijderen
- Voorbeeld: contactgegevens verzamelen om een dienst te leveren

3. Verwerkingsgrondslagen

- AVG: kies een passende grondslag voor het doel
- Toestemming; overeenkomst of precontractuele stappen
- Wettelijke verplichting; vitale belangen
- Taak van algemeen belang of openbaar gezag
- Gerechtvaardigde belangen
- Toestemming is niet altijd vereist of passend

4. Rollen en verantwoordelijkheden

- Verwerkingsverantwoordelijke: bepaalt doel en middelen
- Verwerker: verwerkt persoonsgegevens namens de verantwoordelijke
- Voorbeeld: een organisatie gebruikt een externe dienstverlener voor klantbeheer

5. Rechten van betrokkenen

- AVG-rechten gelden onder voorwaarden en zijn niet absoluut
- Inzage en rectificatie
- Wissing en beperking van verwerking
- Dataportabiliteit en bezwaar

6. Beveiligingsmaatregelen

- Beperk toegang en verzamel niet meer gegevens dan nodig
- Gebruik sterke authenticatie en versleuteling
- Maak back-ups; houd relevante gebeurtenissen bij met logging
- Bereid incidentrespons voor
- Passende maatregelen hangen af van context en risico

7. Datalekken en meldingen

- **Datalek:** ongeoorloofde vernietiging, verlies, wijziging, verstrekking of toegang tot persoonsgegevens
- Niet elk beveiligingsincident is een meldingsplichtig datalek
- AVG: meld bepaalde inbreuken aan de toezichthouder, in beginsel zo mogelijk binnen 72 uur na kennisname, tenzij een risico voor rechten en vrijheden onwaarschijnlijk is
- Bij hoog risico moeten betrokkenen worden geïnformeerd, behoudens toepasselijke uitzonderingen

8. Korte samenvatting

- Bepaal welke persoonsgegevens worden verwerkt en waarom
- Kies een passende AVG-grondslag en bescherm de gegevens
- Ken de rollen, rechten en voorwaarden voor meldingen
- AVG en LGPD zijn vergelijkbare kaders, maar niet identiek
- Onthoud: Dit is een educatieve mindmap, geen juridisch advies.

081. Gegevensclassificatie & Data Governance

1. Wat is gegevensclassificatie?

- Indeling naar gevoeligheid en gevolgen van openbaarmaking, wijziging of verlies.
- Helpt bepalen welke bescherming passend is.

Definities en omgangsregels kunnen per organisate verschillen.

2. De vier niveaus.

Oplopend in gevoeligheid:

- Openbaar — bedoeld voor vrije verspreiding.
- Intern — voor gebruik binnen de organisatie.
- Vertrouwelijk — beperkte toegang.
- Geheim — bekendmaking kan ernstige schade veroorzaken.

3. Voorbeelden.

Illustraties, geen universele classificaties:

- Openbaar gepubliceerde brochure.
- Interne werkinstructie.
- Vertrouwelijke klantgegevens.
- Geheime informatie waarvan bekendmaking ernstige schade kan veroorzaken.

Volg de regels van de eigen organisatie.

4. Dataretentie.

- Bepaal hoelang gegevens worden bewaard.
- Daarna veilig verwijderen of volgens beleid archiveren.
- Bewaartermijnen hangen af van het toepasselijke organisatiebeleid.

5. Eigenaarschap.

- Een aangewezen rol of persoon besluit over classificatie, toegang en bewaartermijn.
- Geen exclusief bezit of onbeperkte toegang.

6. Governance-afspraken.

- Leg beleid en verantwoordelijkheden vast.
- Ken passende toegangsrechten toe.
- Beheer bewaartermijnen en besluiten.

7. Controles en periodieke beoordeling.

- Beoordeel periodiek of toegangsrechten nog passend zijn.
- Leg besluiten en beoordelingen vast.
- Controleer naleving van classificatie- en retentieafspraken.

8. Korte samenvatting.

- Classificatie stuurt bescherming.
- Eigenaarschap ondersteunt besluiten.
- Retentie begrenst bewaring.
- Governance organiseert naleving.

Bedrijfscontinuïteit (BCP) & Disaster Recovery (DRP)

1. Wat is het

- **BCP**: essentiële activiteiten voortzetten tijdens en na een verstoring.
- **DRP**: IT-systemen en gegevens herstellen na een ernstige verstoring.
- BCP omvat mensen, processen en voorzieningen; DRP ondersteunt dit met IT-herstel.

2. Waarom is het belangrijk

- Beperk de gevolgen van uitval voor essentiële activiteiten.
- Maak herstelprioriteiten en verantwoordelijkheden vooraf duidelijk.
- Stem maatregelen af op de bedrijfsimpact en hersteldoelen.

3. RPO en RTO

Illustratief voorbeeld

- **RTO**: beoogde maximale hersteltijd na uitval.
- **RPO**: maximaal aanvaardbare periode aan gegevensverlies, gerekend terug vanaf de verstoring.
- Voorbeeld: RTO 4 uur betekent herstel binnen 4 uur; RPO 1 uur betekent hoogstens het laatste uur aan wijzigingen verliezen.
- Dit zijn illustratieve doelen, geen universele normen.

4. Redundantie

- Reserveonderdelen of alternatieve voorzieningen vangen een storing mogelijk op.
- Voorbeelden: tweede netwerkverbinding, redundante opslag of uitwijkomgeving.
- Controleer afhankelijkheden en hoe overschakeling werkt; redundantie alleen garandeert geen herstel.

5. Herstelstrategieën

- Maak back-ups en test of gegevens daaruit te herstellen zijn.
- Leg herstelprocedures vast en overweeg alternatieve infrastructuur.
- Herstel kritieke diensten volgens bedrijfsimpact en vastgestelde doelen.

6. Rollen en communicatie

- Wijs verantwoordelijkheden voor besluitvorming en herstel toe.
- Spreek af wie betrokkenen informeert, wanneer en via welke kanalen.
- Neem relevante teams, processen en voorzieningen mee in de afspraken.

7. Testen en verbeteren

- Oefen periodiek de continuïteits- en herstelafspraken.
- Controleer procedures, back-ups, afhankelijkheden en communicatie.
- Werk plannen bij op basis van bevindingen en gewijzigde hersteldoelen.

8. Korte samenvatting

- BCP helpt essentiële activiteiten voort te zetten; DRP richt zich op herstel van IT en gegevens.
- RTO gaat over hersteltijd; RPO over aanvaardbaar gegevensverlies.
- Geteste maatregelen, heldere rollen en oefeningen ondersteunen herstel, maar bieden geen garantie.

Veilige back-upstrategieën & onveranderlijkheid

1. Wat is het?

Een back-up is een extra kopie van gegevens die je kunt gebruiken om ze te herstellen. Een robuuste aanpak bewaart meerdere kopieën en houdt rekening met opslag, toegang en herstel.

- Bewaar kopieën op verschillende opslagmedia of locaties.
- Zorg dat niet elke kopie voortdurend bereikbaar is vanaf het gewone netwerk.

2. Waarom belangrijk

Bij verlies, schade of versleuteling van gegevens kan een bruikbare back-up helpen om werk te herstellen. Back-ups beperken mogelijke gevolgen, maar voorkomen niet vanzelf een ransomware-infectie.

- Een kopie die ook bereikbaar is voor een aanvaller kan worden geraakt.
- Een hersteltest helpt vaststellen of gegevens daadwerkelijk terug te zetten zijn.

3. 3-2-1-1-0-regel

Gebruik deze regel als praktische vuistregel, niet als garantie:

- 3 kopieën van je gegevens.
- 2 verschillende opslagmedia.
- 1 kopie buiten de primaire locatie.
- 1 kopie offline of onveranderlijk.
- 0 fouten bij een gecontroloerde hersteltest.

Voorbeeld

Bewaar de originele bestanden op een computer, een back-up op een extern opslagmedium en een derde kopie op een andere locatie. Houd één kopie offline of onveranderlijk en test regelmatig of je bestanden kunt herstellen.

4. Air-gapping

Een air gap is een fysieke of logisch afgedwongen scheiding tussen een back-upomgeving en de gewone netwerkverbindingen. Offline opslag is niet continu bereikbaar vanaf het productienetwerk.

- Scheiding kan de bereikbaarheid voor aanvallers verkleinen.
- Een offline kopie moet nog steeds zorgvuldig worden beheerd en op herstelbaarheid getest.

5. Onveranderlijke back-ups

Een onveranderlijke (immutable) back-up kan gedurende een ingestelde bewaartermijn niet worden gewijzigd of verwijderd via gewone beheerhandelingen.

- Dit kan helpen beschermen tegen ransomware die bereikbare back-ups probeert te versleutelen of te wissen.
- De bescherming geldt binnen de ingestelde voorwaarden en vervangt geen hersteltest.

6. Ransomwarebescherming

Ransomware kan bestanden op bereikbare systemen versleutelen. Back-ups die gescheiden zijn van die systemen en periodiek op herstelbaarheid worden getest, bieden een mogelijke herstelroute.

- Een back-up voorkomt op zichzelf geen infectie.
- Scheiding en hersteltests kunnen helpen de gevolgen van gegevensversleuteling te beperken.

7. Praktische aandachtspunten

Stem de back-upaanpak af op de gegevens en de omgeving. Denk niet alleen aan het maken van kopieën, maar ook aan bescherming en herstel.

- Beperk en beveilig de toegang tot back-ups.
- Kies passende bewaartermijnen en opslaglocaties.
- Test periodiek of gegevens correct kunnen worden hersteld.
- Controleer of offline of onveranderlijke kopieën beschikbaar blijven wanneer nodig.

8. Korte samenvatting

Meerdere kopieën, opslag op verschillende media of locaties, een offline of onveranderlijke kopie en een hersteltest vormen samen een praktische basis voor gegevensherstel.

- Zie 3-2-1-1-0 als vuistregel, niet als garantie.
- Een goed plan helpt herstellen; het voorkomt niet vanzelf een aanval.

Beveiliging van het Internet der Dingen (IoT)

1. Wat is IoT-beveiliging?

- Internet der Dingen: apparaten die via een netwerk gegevens uitwisselen.
- Sensoren meten; actuatoren voeren opdrachten uit; ingebouwde rekenfuncties verwerken gegevens.
- Voorbeelden: slimme camera, thermostaat en industriële sensor.

2. Kwetsbaarheden

- Standaard- of zwakke wachtwoorden en gebrekkig toegangsbeheer.
- Verouderde firmware en onnodige netwerkdiensten.
- Onveilige opslag of onvoldoende versleuteling van gegevens.

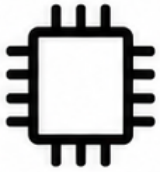
3. Protocollen

- IP/TCP en UDP ondersteunen netwerkcommunicatie.
- HTTP/HTTPS voor webverkeer; MQTT en CoAP voor berichtenuitwisseling tussen apparaten.
- Wifi, Bluetooth Low Energy en Zigbee zijn veelgebruikte draadloze technologieën.
- Welke techniek past, hangt af van apparaat en toepassing; een protocol garandeert op zichzelf geen veiligheid.

4. Dreigingen en gevolgen

- Ongewenste bediening van een apparaat.
- Verlies van vertrouwelijkheid of integriteit van gegevens.
- Verstoring van de beschikbaarheid.
- Een kwetsbaar apparaat kan als springplank naar andere systemen worden misbruikt.

5. Apparaten en ingebedde hardware

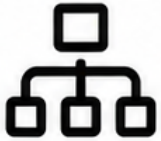


- Beperk fysieke toegang tot apparaten en beheerinterfaces.
- Debugpoorten kunnen risico geven als ze onnodig actief of onbeveiligd zijn.
- Bescherm firmware en opgeslagen sleutels; schakel onnodige interfaces uit of scherm ze af.

6. Hardening

- Vervang standaardgegevens door unieke, sterke inloggegevens.
- Pas firmware tijdig bij via een vertrouwde bron.
- Schakel ongebruikte functies en diensten uit.
- Gebruik versleutelde communicatie waar het apparaat dit ondersteunt en beperk toegang tot beheerinterfaces.

7. Netwerk en onderhoud



- Scheid IoT-apparaten waar mogelijk van vertrouwde gebruikers- of bedrijfsnetwerken.
- Beperk inkomende en uitgaande verbindingen tot wat noodzakelijk is.
- Houd apparaten en configuraties bij; controleer of updates beschikbaar zijn.
- Let op onverwachte verbindingen of afwijkend gedrag; vervang of verwijder apparaten veilig wanneer ze niet meer nodig zijn.

8. Korte samenvatting

- Ken je apparaten en hun netwerkverbindingen.
- Verklein risico's met sterke gegevens, updates, minimale functies en passende versleuteling.
- Segmenteer het netwerk en volg apparaten op als onderdeel van periodiek onderhoud.
- Dit is een educatief overzicht, geen diagnose of professioneel beveiligingsadvies.

Draadloze aanvalstechnieken: Rogue AP, Evil Twin & KRACK

1. Wat is het?

- Draadloze aanvallen richten zich op wifi-toegang en communicatie.
- Een ongewenst toegangspunt kan een extra toegangspad tot een netwerk vormen.
- **Rogue AP**, Evil Twin en KRACK zijn verschillende risico's; de verdediging verschilt per type.

2. Rogue AP

- Een niet-goedgekeurd draadloos toegangspunt dat op een netwerk is aangesloten of toegang daartoe biedt.
- Kan beveiligingsbeleid en centraal beheer omzeilen.
- **Voorbeeld:** een onbekend toegangspunt verschijnt in een kantooromgeving.

3. Evil Twin

- Een kwaadwillig toegangspunt dat zich voordoet als een vertrouwd wifi-netwerk.
- Gebruikt vaak een vergelijkbare netwerknaam (SSID) om gebruikers te misleiden.
- Kan verkeer helpen observeren of gebruikers naar nagemakte diensten leiden.

4. KRACK

- Afkorting van Key Reinstallation Attack.
- De in 2017 beschreven aanval benut zwakheden in de WPA2-handshake om een al gebruikte cryptografische sleutel opnieuw te installeren.
- De mogelijke gevolgen hangen af van de protocolimplementatie en het apparaat; beveiligingsupdates zijn belangrijk.

5. Aanvalsketen

- Een aanvaller creëert of benut een ongewenst toegangspunt.
- Een gebruiker maakt verbinding of een kwetsbare handshake wordt misbruikt.
- Verkeer of netwerktoegang kan risico lopen; de precieze impact verschilt per aanval en configuratie.
- Versleuteling en correcte controle van servercertificaten helpen beschermen tegen meekijken en imitatie.

6. Herkennen

- Let op onbekende toegangspunten en afwijkende SSID/BSSID-combinaties.
- Controleer bij twijfel de netwerknaam via een vertrouwd kanaal; vertrouw niet alleen op een bekende naam.
- Neem waarschuwingen over certificaatfouten serieus en meld verdachte wifi aan de netwerkbeheerder.

Voorzichtig: Verbreek de verbinding met een verdacht netwerk en meld het volgens de afspraken van je organisatie.

7. Tegenmaatregelen

- Installeer beveiligingsupdates op toegangspunten en apparaten.
- Gebruik **WPA3** waar dit wordt ondersteund, of goed geconfigureerde WPA2-AES.
- Schakel automatisch verbinden met ongebruikte wifi-netwerken uit.
- **Beheerders:** inventariseer goedgekeurde toegangspunten, controleer op onbekende radio's en segmenteer gastnetwerken.
- Gebruik waar passend 802.1X met certificaatgebaseerde authenticatie.
- Op openbare wifi: gebruik HTTPS, deel geen gevoelige gegevens via onbekende netwerken en negeer geen certificaatwaarschuwingen. Een VPN kan extra bescherming bieden, maar maakt een onbetrouwbaar netwerk niet vanzelf veilig.

8. Korte samenvatting

- **Rogue AP:** ongeautoriseerd toegangspunt; beheer en inventarisatie zijn belangrijk.
- **Evil Twin:** imitatie van een vertrouwd wifi-netwerk; controleer identiteit en certificaten.
- **KRACK:** aanval op sleutelinstallatie in WPA2; houd apparaten bijgewerkt.
- **Bij twijfel:** verbreek de verbinding, meld het incident en gebruik een vertrouwd netwerk.

API-beveiliging & OWASP API Security Top 10

Bescherm API's door identiteit, rechten, gegevens en gebruik serverzijdig te controleren.

1. Wat is het

- **API:** koppeling waarmee software gegevens of functies opvraagt.
- **REST** en **GraphQL** zijn twee veelgebruikte API-stijlen.
- **OWASP API Security Top 10:** aandacht voor belangrijke API-beveiligingsrisico's.

2. API-aanvalsoppervlak

- Eindpunten, acties en gegevens kunnen doelwit zijn.
- **REST:** controleer ieder eindpunt en object.
- **GraphQL:** één verzoek kan meerdere velden en geneste relaties opvragen.
- Beperk querydiepte, complexiteit en verzoekfrequentie waar passend.

3. BOLA en BFLA

- **BOLA:** een gebruiker krijgt zonder voldoende controle toegang tot een specifiek object.
- **Voorbeeld:** wijzig een order-ID en bekijk zo een bestelling van iemand anders.
- **BFLA:** een gebruiker kan een functie aanroepen waarvoor die geen rechten heeft.
- **Voorbeeld:** een gewone gebruiker roept een beheerdersactie aan.

Controleer per verzoek
Controleer serverzijdig of deze gebruiker toegang heeft tot dit object én deze actie mag uitvoeren.

4. Authenticatie en autorisatie

- Authenticatie: wie doet het verzoek?
- Autorisatie: wat mag die gebruiker doen?
- Een geldige login of token geeft niet automatisch toegang tot ieder object of elke functie.
- Controleer rechten serverzijdig voor gevoelige acties.

5. Gegevensblootstelling en massatoewijzing

- Geef alleen responsvelden terug die de toepassing nodig heeft.
- Verberg gevoelige gegevens niet alleen in de gebruikersinterface.
- Sta bij wijzigingen alleen expliciet toegestane invoervelden toe.
- Controleer gevoelige eigenschappen serverzijdig.

6. Rate limiting en resourcegebruik

- Beperk het aantal of tempo van verzoeken.
- Stel limieten in per account, token, IP-adres of functie waar passend.
- Dit kan misbruik en uitputting van middelen helpen beperken.
- Rate limiting vervangt geen authenticatie, autorisatie of invoervalidatie.

7. Veilige ontwerp- en testmaatregelen

- Voer beveiligingscontroles serverzijdig uit.
- Test in een omgeving waarvoor je toestemming hebt.
- Positieve test: toegestane toegang werkt.
- Negatieve test: toegang tot andermans objecten of verboden functies wordt geweigerd.

8. Korte samenvatting

- Controleer identiteit én rechten.
- Controleer toegang per object en per functie.
- Beperk gegevens en verzoekverbruik.
- Test toegestane én geweigerde toegang.

Mindmap 089: Veilige tokenauthenticatie (JWT, OAuth 2.0 en OIDC)

1. Wat is het?

- Een token is een digitaal bewijs dat een toepassing kan gebruiken binnen een beveiligingsproces.
- Authenticatie beantwoordt: wie ben je? Autorisatie bepaalt: waartoe heb je toegang.
- JWT is een compact tokenformaat; OAuth 2.0 en OIDC hebben andere, aanvullende rollen.

2. OAuth 2.0

- Een raamwerk voor autorisatie: een client krijgt beperkte toegang tot een beschermde bron.
- Toegang kan namens een gebruiker of namens de client zelf worden aangevraagd.
- OAuth 2.0 alleen is geen protocol voor gebruikersauthenticatie.

3. OpenID Connect

- OIDC voegt een authenticatielaag toe aan OAuth 2.0.
- Een ID-token geeft de client informatie over een authenticatiegebeurtenis.
- Een access-token is bedoeld voor toegang tot een beschermde bron, het is geen ID-token.

4. JWT-opbouw

- Compacte structuur met doorgaans een header, payload en handtekening of MAC.
- De payload bevat claims, bijvoorbeeld over de uitgever of geldigheidsduur.
- Codering is geen versleuteling: de inhoud van een JWT kan leesbaar zijn.

5. Tokenvalidatie

- Controleer de cryptografische integriteit met vertrouwde sleutels en toegestane algoritmen.
- Controleer relevante claims en context, zoals uitgever, doelgroep en vervaltijd.
- Verwerp een verlopen token of een token voor de verkeerde doelgroep.
- Vertrouw claims niet voordat het token correct is gevalideerd.

6. Tokenkaping

- Wie een bearer-token bezit, kan het doorgaans gebruiken; blootstelling kan misbruik mogelijk maken.
- Onveilige opslag, logs, URL's of onbeveiligde overdracht vergroten het risico op diefstal of hergebruik.
- Deel tokens niet en neem ze niet op in logs of URL's.

7. Bescherming

- Gebruik TLS en beperk waar tokens terechtkomen; sla ze zorgvuldig op.
- Beperk de geldigheidsduur en gebruik waar passend intrekking of rotatie.
- Gebruik bij browserflows waar passend de autorisatiecodestroom met PKCE; valideer omleidings-URL's en gebruik state- en nonce-controles.

Onthoud | Een gedecodeerd token is nog niet vertrouwd. Valideer de integriteit én de context voordat je het gebruikt.

8. Korte samenvatting

- OAuth 2.0 regelt autorisatie; OIDC voegt authenticatie toe; JWT is een tokenformaat.
- ID-token en access-token zijn niet uitwisselbaar.
- Valideer tokens, controleer hun context en bescherm ze tegen blootstelling.

Cryptografische protocollen & sleutelbeheer (KMS/HSM)

1. Wat is het?

- **Afspraken** voor veilige cryptografische bewerkingen en berichtenuitwisseling.
- **Sleutelbeheer** beschermt sleutels gedurende hun hele levenscyclus.

2. Cryptografische bouwstenen

- **Symmetrisch:** dezelfde geheime sleutel versleutelt en ontsleutelt.
- **Asymmetrisch:** een sleutelpaar met publieke en private sleutel.
- **Hashfunctie:** maakt een vaste vingerafdruk; is geen versleuteling en geeft op zichzelf geen geheimhouding.

3. Protocol en doelen

- Beschrijft hoe partijen bewerkingen en berichten uitwisselen.
- **Doelen:** vertrouwelijkheid, integriteit, authenticatie of sleutelovereenkomst.
- **Voorbeeld:** TLS beveiligt netwerkverbindingen.

4. Sleutellevenscyclus

- Genereren en beschikbaar stellen.
- Toegang beperken en sleutels veilig gebruiken.
- Back-up waar passend; roteren volgens beleid.
- Bij einde of risico: intrekken en vernietigen.
- Bescherm sleutels tegen ongeautoriseerde inzage en wijziging.

5. KMS

- Een sleutelbeheersysteem beheert sleutels en gebruiksbeleid.
- Controleert identiteit en toegestane handelingen.
- Houdt bij welke sleutelhandelingen plaatsvinden.

6. HSM

- Een hardwarebeveiligingsmodule beschermt sleutels en kan cryptografische bewerkingen uitvoeren.
- Kan onderdeel zijn van een KMS-oplossing.
- HSM en KMS zijn niet hetzelfde.

7. Beveiligingsmaatregelen

- Pas het minsteprivilegebeginsel toe: alleen noodzakelijke toegang.
- Scheid beheerrollen waar passend en registreer sleutelgebruik.
- Bescherm communicatie tussen toepassingen en sleutelvoorzieningen.
- Rotatie vervangt sleutels, maar maakt eerdere blootstelling niet ongedaan.

8. Korte samenvatting

- Protocollen regelen veilige cryptografische uitwisseling; sleutelbeheer beschermt de sleutels.
- KMS beheert sleutels en beleid; HSM biedt gespecialiseerde hardwarebescherming.
- **Voorbeeld:** een toepassing vraagt via een KMS versleuteling aan. De policy controleert identiteit en handeling; een HSM kan de bewerking uitvoeren zonder de beschermde sleutel als leesbare waarde vrij te geven.

Postkwantumcryptografie & toekomstige dreigingen

1 Wat is kwantumcomputing?

Een kwantumcomputer verwerkt informatie met qubits. Voor sommige wiskundige problemen biedt dat een andere aanpak dan een klassieke computer. Dat betekent niet dat een kwantumcomputer alle taken sneller uitvoert.

- Qubits zijn informatie-eenheden van kwantumcomputers.
- Gevolgen hangen af van probleem en algoritme.
- Een praktisch inzetbare kwantumcomputer die huidige cryptografie op grote schaal bedreigt, is geen gegeven.

Belangrijk onderscheid — Een algoritmische mogelijkheid is niet hetzelfde als een bestaande computer die dit op cryptografisch relevante schaal kan uitvoeren.

2 Asymmetrische cryptografie

Gebruikt sleutelpaar: publieke sleutel en geheime privésleutel. Veel gangbare systemen steunen op wiskundige problemen die klassiek moeilijk zijn.

- RSA: moeilijkheid van factorisatie van grote getallen.
- Elliptische krommen: problemen rond discrete logaritmen.
- Ondersteunt onder meer sleuteluitwisseling en digitale handtekeningen.

3 De kwantumdreiging: Shor en Grover

Een voldoende krachtige, foutgecorrigeerde kwantumcomputer zou met Shors algoritme gebruikelijke RSA- en elliptische-krommesystemen kunnen bedreigen. Grovers algoritme versnelt bepaalde zoekproblemen en heeft een ander effect.

- Shor richt zich op factorisatie en discrete logaritmen; relevant voor RSA en elliptische krommen.
- Grover is relevant voor zoekproblemen, waaronder toepassingen bij symmetrische cryptografie en hashfuncties.
- Gevolgen verschillen per algoritme; niet alle encryptie is onmiddellijk gebroken.

Geen reden tot paniek — Dreiging hangt af van toekomstige hardwarecapaciteit. Shor en Grover hebben niet hetzelfde effect op cryptografie.

4 Wat is postkwantumcryptografie?

Postkwantumcryptografie (PQC) omvat algoritmen bedoeld bestand te zijn tegen aanvallen met klassieke én kwantumcomputers.

- PQC draait op conventionele computers; zelf geen kwantumcomputer nodig.
- PQC is iets anders dan kwantumcryptografie.
- Doel: bescherming behouden wanneer kwantumcomputing zich verder ontwikkelt.

5 Eerste NIST-standaarden

In 2024 publiceerde NIST drie eerste standaarden voor postkwantumcryptografie: één voor sleutelinkapseling en twee voor digitale handtekeningen.

- **FIPS 203** — ML-KEM (gebaseerd op CRYSTALS-Kyber): sleutelinkapseling.
- **FIPS 204** — ML-DSA (gebaseerd op CRYSTALS-Dilithium): digitale handtekeningen.
- **FIPS 205** — SLH-DSA (gebaseerd op SPHINCS+): digitale handtekeningen.

Onthoud het verschil — ML-KEM is voor sleutelinkapseling. ML-DSA en SLH-DSA zijn voor digitale handtekeningen.

6 Risico's en voorbeelden

Langetermijnrisico: 'nu onderscheppen, later ontsleutelen': iemand verzamelt versleutelde communicatie nu, in de hoop die later te ontsleutelen wanneer benodigde kwantumcapaciteit beschikbaar is.

- Gegevens die lang vertrouwelijk moeten blijven verdienen aandacht.
- Dit betekent niet dat verzamelde communicatie nu al ontsleuteld kan worden.
- Niet alle gegevens of cryptografische toepassingen lopen hetzelfde risico.

7 De transitie

Overgang naar PQC vraagt inzicht in huidig cryptografiegebruik en een gecontroleerde aanpak. Aandachtspunten zijn algemeen educatief, geen organisatie-specifiek advies.

- Breng in kaart waar cryptografie wordt gebruikt.
- Bepaal welke systemen en gegevens langdurige vertrouwelijkheid vereisen.
- Volg toepasselijke standaarden en plan migratie zorgvuldig.
- Test interoperabiliteit: kunnen systemen na de overgang veilig communiceren?

8 Korte samenvatting

Kwantumcomputing kan specifiede cryptografische systemen onder druk zetten, maar niet alle cryptografie op dezelfde manier. Postkwantumcryptografie biedt gestandaardiseerde algoritmen voor gewone computers.

- Shor bedreigt met passende toekomstige hardware RSA en elliptische krommen; Grover versnelt bepaalde zoekproblemen.
- NIST publiceerde in 2024 ML-KEM voor sleutelinkapseling en ML-DSA en SLH-DSA voor handtekeningen.
- Voorbereiding begint met inzicht in cryptografiegebruik, relevante gegevens en gecontroleerde migratie.

Reverse engineering: grondbeginselen van software

1. Wat is het

- Onderzoek hoe software is opgebouwd en werkt.
- Analyseer binaries: uitvoerbare bestanden of andere machinecode.
- Doel: gedrag begrijpen en aanwijzingen vinden voor malwareonderzoek.

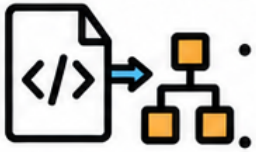
2. Statische analyse

- Bestudeer de binary zonder deze uit te voeren.
- Bekijk bestandstype en metadata; inspecteer leesbare tekstfragmenten (strings).
- Disassembleer instructies om aanwijzingen in de code te zoeken.

3. Dynamische analyse

- Observeer het programma tijdens uitvoering in een geïsoleerde testomgeving.
- Let op activiteit rond bestanden, processen en netwerkverbindingen.
- Vergelijk het waargenomen gedrag met hypotheses uit de statische analyse.

4. Decompilatie



- Zet machinecode of bytecode om naar een representatie op hoger niveau.
- Gebruik de uitkomst om de programmastructuur en mogelijke logica te begrijpen.
- De reconstructie is een benadering, niet noodzakelijk de oorspronkelijke broncode.

5. Disassembler

```
MOV  EAX, EBX
ADD  EAX, 0x10
JMP  0x004010
```

- Zet machinecode om in assembly-instructies.
- Helpt instructies en codepaden stap voor stap te bekijken.
- De uitvoer is lager-niveau dan de representatie van een decompiler.

6. Debugger

- Laat een programma stapsgewijs uitvoeren en pauzeren.
- Inspecteer tijdens uitvoering onder meer registers en geheugen.
- Gebruik observaties om te begrijpen wat code op een bepaald moment doet.

7. Veilige werkwijze

- Werk in een geïsoleerde, gecontroleerde testomgeving.
- Begin met inspectie waarbij het bestand niet wordt uitgevoerd.
- Voer uitsluitend geautoriseerde monsters gecontroleerd uit.
- Leg waarnemingen vast en verspreid malware niet.

Veiligheid eerst
Voer onbekende of verdachte bestanden niet uit op een gewone computer of netwerk.

8. Korte samenvatting

- Statisch: code onderzoeken zonder uitvoering.
- Dynamisch: gedrag observeren tijdens gecontroleerde uitvoering.
- Disassembler en debugger tonen verschillende aspecten van de code; decompiler geeft een benadering op hoger niveau.
- Combineer bevindingen zorgvuldig en werk alleen geautoriseerd en geïsoleerd.

Softwarebeveiliging: fuzzing en buffer-overflowgrondbeginselen

1. Wat is geheugencorruptie?

Geheugencorruptie ontstaat wanneer geheugen dat bij een programma hoort onbedoeld wordt gewijzigd of overschreven.

- Een buffer overflow treedt op wanneer gegevens buiten de grenzen van een buffer worden geschreven.
- Gevolgen kunnen variëren van beschadigde gegevens tot een crash.

2. Buffergrenzen

Een buffer is een geheugenruimte voor gegevens. Schrijven buiten de toegewezen grenzen kan aangrenzende gegevens aantasten.

- Denk aan een invoerveld dat een langere tekst ontvangt dan de buffer aankan.
- De exacte gevolgen hangen af van het programma en de omgeving.

3. Stack-overflow

Een buffer op de stack kan over zijn grenzen worden beschreven; naburige stackgegevens kunnen daardoor worden aangetast.

- Dat kan lokale variabelen of besturingsgegevens raken.
- Stack overflow kan óók betekenen dat de stackruimte uitgeput raakt, bijvoorbeeld door te diepe aanroepen. Dat is een ander verschijnsel.

4. Heap-overflow

Een heap overflow is een overschrijving buiten de grenzen van een buffer die dynamisch op de heap is toegewezen.

- Mogelijke gevolgen zijn beschadigde gegevens of een crash.
- Dit is niet hetzelfde als een buffer op de stack die buiten zijn grenzen wordt beschreven.

5. Wat is fuzzing?

Fuzzing is geautomatiseerd testen waarbij een programma veel verschillende invoeren krijgt om onverwacht gedrag, fouten of crashes te vinden.

- Voorbeelden: een eigen testprogramma met een invoerveld of een parser voor onverwachte bestandsinvoer.
- Gebruik uitsluitend eigen of expliciet geautoriseerde testdoelen.

Belangrijk
Een crash kan wijzen op een foutpad, maar bewijst op zichzelf niet dat er een exploiteerbare kwetsbaarheid is.

6. Fuzzer-methoden

Een fuzzer kan invoer op verschillende manieren samenstellen of aanpassen.

- Willekeurig: genereert uiteenlopende invoer.
- Mutatiegebaseerd: past bestaande testinvoer aan.
- Model- of grammaticagebaseerd: maakt invoer volgens bekende structuren.
- Code coverage-fuzzing gebruikt feedback over bereikte codepaden om volgende invoer te sturen.

7. Crashes onderzoeken

Een afgebakende, toegestane testcyclus helpt bevindingen reproduceerbaar en bruikbaar te maken.

- Baken het testdoel af en gebruik een geautoriseerde testomgeving.
- Genereer of muteer invoer; voer het programma uit en observeer het gedrag.
- Bewaar crashes en herhaal de test om te controleren of ze terugkomen.
- Verklein reproduceerbare invoer tot een minimale testcase voor triage en foutanalyse.

8. Korte samenvatting

Buffer overflows zijn schrijffacties buiten buffergrenzen; fuzzing helpt zulke fouten systematisch op te sporen.

- Stack en heap verwijzen naar verschillende soorten geheugenruimte; onderscheid een stack-bufferoverschrijding van stackuitputting.
- Een crash is een aanwijzing om te onderzoeken, geen bewijs van exploiteerbaarheid.
- Test alleen met toestemming; reproduceer en minimaliseer bevindingen.

Netwerkforensica & PCAP-analyse

1. Wat is netwerkforensica?

Onderzoek van vastgelegd netwerkverkeer om gebeurtenissen en communicatie-patronen beter te begrijpen.

- Bekijk wat er tussen systemen is uitgewisseld.
- Gebruik waarnemingen als aanwijzingen, niet als bewijs op zichzelf.

2. PCAP en capturecontext

Een PCAP-bestand bevat vastgelegde netwerkpakketten die achteraf kunnen worden geïnspecteerd.

- Noteer bron, tijdsbereik en omstandigheden van de opname.
- Conclusies zijn begrensd door wat tijdens de opname zichtbaar en vastgelegd is.
- Ontbrekende pakketten kunnen het beeld onvolledig maken.

3. Verkeersoverzicht

Begin met een breed overzicht voordat je afzonderlijke gesprekken onderzoekt.

- Inventariseer protocollen en eindpunten.
- Let op terugkerende verbindingen en onverwachte bestemmingen.
- Een opvallend patroon vraagt om context; het bewijst geen kwaadwillend gedrag.

4. Wireshark-filters

Displayfilters helpen pakketten te selecteren op eigenschappen. Ze veranderen de opgeslagen capture niet.

- **dns** — selecteer DNS-verkeer.
- **tcp.port == 443** — selecteer TCP-verkeer op poort 443.
- **ip.addr == 192.0.2.10** — selecteer verkeer met dit IP-adres.

5. Stroomanalyse

TCP-stroomanalyse groepeert pakketten van een verbinding en helpt de volgorde en inhoud van communicaite te bekijken.

- Volg een stroom alleen als de benodigde pakketten in de capture aanwezig zijn.
- Versleutelde payload is zonder passende ontsleuteling geen leesbare applicatie-inhoud.

6. C2-signalen

Een command-and-controlkanaal (C2) kan een gecompromitteerd systeem met een besturings-infrastructuur verbinden voor het ontvangen van opdrachten of uitwisselen van gegevens.

- Mogelijke signalen: ongebruikelijk periodieke verbindingen, verdachte DNS-verzoeken, onverwachte bestemmingen of afwijkende protocolpatronen.
- Geen enkel signaal bewijst op zichzelf kwaadaardige activiteit.

Onderzoek met context
Vergelijk signalen met de capturecontext en andere beschikbare informatie; noteer onzekerheden.

7. Onderzoeksworkflow

Werk stapsgewijs en leg waarnemingen én beperkingen vast.

1. **Context bepalen**
   Noteer de herkomst van de capture en het vastgelegde tijdsbereik.
2. **Overzicht maken**
   Inventariseer protocollen en eindpunten.
3. **Gericht filteren**
   Pas displayfilters toe om relevante pakketten te selecteren.
4. **Stromen inspecteren**
   Bekijk relevante verbindingen voor zover de capture voldoende pakketten bevat.
5. **Bevindingen vastleggen**
   Noteer observaties, onzekerheden en beperkingen; behandel bevindingen als aanwijzingen die aanvullende context vereisen.

8. Korte samenvatting

PCAP-analyse helpt netwerkverkeer achteraf te onderzoeken, maar alleen binnen de grenzen van de opname.

- Begin bij context en overzicht; zoom daarna gericht in met filters en stroomanalyse.
- Beschouw mogelijke C2-signalen als aanwijzingen, niet als sluitend bewijs.
- Houd rekening met ontbrekende pakketten en versleutelde inhoud.

E-mailforensica en phishinganalyse

1. Wat is het

- Verdachte e-mail veilig en technisch beoordelen.
- Phishing probeert je bijvoorbeeld tot klikken, betalen of inloggen te verleiden.
- Eén opvallend signaal bewijst geen fraude; bekijk de context.

2. E-mailopbouw

- Bericht: afzender, onderwerp, inhoud en eventuele bijlagen.
- Technische headers bevatten informatie over verzending en opmaak.
- Een zichtbare afzendernaam is op zichzelf geen bewijs van identiteit.

3. MIME-headers

- MIME structureert inhoud en bijlagen als onderdelen met eigen inhoudstypen en codering.
- Multipart-berichten combineren bijvoorbeeld tekst en een bijlage.
- Base64 is een overdrachtscodering, geen versleuteling.

4. Received-sporen

- Een bericht kan meerdere Received-regels bevatten; mailservers voegen doorgaans hun eigen transportregel toe.
- Lees de keten als mogelijke route van onder naar boven.
- Aangeleverde headers bewijzen op zichzelf niet wie de afzender is of dat het bericht authentiek is.

5. Bijlagen veilig beoordelen

- Open geen onverwachte bestanden.
- Noteer bestandsnaam, extensie en berichtcontext; naam en extensie bewijzen het bestandstype niet.
- Analyse hoort alleen in een geïsoleerde, toegestane omgeving en volgens lokale procedures.

6. URL's defangen

- Maak een link minder direct aanklikbaar door bijvoorbeeld https te schrijven als hxxps en punten als [.] weer te geven.
- Voorbeeld: hxxps://voorbeeld[.]nl/pagina.
- Dit verlaagt linkrisico bij delen of documenteren, maar verwijdert geen malware en garandeert geen veiligheid.

7. Analyse en veilig handelen

- Let op onverwachte urgentie, een afwijkend domein, een ongebruikelijke betaal- of inlogvraag en een onverwachte bijlage.
- Klik niet, antwoord niet en voer geen bijlage uit.
- Verifieer verzoeken via een al bekende contactroute en meld het bericht volgens de organisatieprocedure.
- Bewaar het bericht bij twijfel volgens die procedure voor bevoegde analyse.

8. Korte samenvatting

- Bekijk inhoud, headers, bijlagen en links in hun context.
- Een aanwijzing is geen bewijs; combineer meerdere signalen.
- Deel links alleen gedefangd en volg de veilige meldprocedure.

Hostforensica: Windows-artefacten

1 Wat is het

Windows-artefacten zijn gegevens die het besturingssysteem en toepassingen achterlaten. Ze kunnen aanwijzingen geven over systeem- en bestandsactiviteit, maar één artefact vertelt zelden het hele verhaal.

- Onderzoekers vergelijken verschillende bronnen en tijdstippen.
- Een aanwijzing is niet automatisch bewijs van wie iets deed of waarom.

Kernidee
Interpreteer artefacten altijd in hun context en combineer ze waar mogelijk.

2 Prefetch

Prefetch is een Windows-mechanisme dat informatie over programmastarts vastlegt om het opstarten te helpen optimaliseren.

- Prefetch-bestanden kunnen aanwijzingen geven dat een programma is gestart.
- Ze bewijzen op zichzelf niet wie het startte of met welke bedoeling.

3 Shimcache en Amcache

Shimcache (AppCompatCache) en Amcache zijn Windows-artefacten die verband houden met compatibiliteit en inventarisatie. Hun gegevens kunnen helpen bestanden of toepassingen te identificeren die het systeem heeft waargenomen.

- Shimcache-vermeldingen bewijzen op zichzelf niet dat een programma is uitgevoerd.
- Amcache kan metadata over toepassingen en bestanden bevatten; aanwezigheid alleen bewijst geen uitvoering.

Let op
Beoordeel deze vermeldingen samen met andere artefacten en de onderzoeksvraag.

4 MFT

De Master File Table (MFT) is de centrale metadatastructuur van een NTFS-volume. Bestandsrecords kunnen informatie bevatten die helpt bij het reconstrueren van bestandsactiviteiten.

- Records kunnen namen, tijdstempels, kenmerken en verwijzingen bevatten.
- Tijdstempels moeten zorgvuldig worden geïnterpreteerd; ze vertellen niet zonder meer het volledige verhaal.

5 Registerhives

Windows-registerhives zijn bestanden met groepen configuratie- en statusgegevens. Welke hive relevant is, hangt af van de onderzoeksvraag.

- Voorbeelden: SYSTEM, SOFTWARE, SAM en SECURITY.
- NTUSER.DAT bevat gegevens die bij een gebruikersprofiel horen.
- Een hive is een gegevensbron; de betekenis van een vondst vraagt om context.

6 Event Logs

Windows Event Logs zijn gestructureerde registraties van gebeurtenissen van het besturingssysteem en toepassingen.

- De inhoud hangt af van ingeschakelde registratie en bewaartermijnen.
- Logs kunnen onvolledig zijn of gewijzigd zijn; afwezigheid van een gebeurtenis is niet doorslaggevend.

7 Samenhang en beperkingen

Een betrouwbare analyse legt artefacten naast elkaar en vergelijkt hun tijdlijnen met de context van het systeem. Zo kunnen bronnen elkaar aanvullen en tegenstrijdigheden zichtbaar maken.

- Vergelijk bijvoorbeeld Prefetch, MFT, registergegevens en Event Logs.
- Controleer welke gegevensbron de aanwijzing opleverde en wat die bron wel en niet kan aantonen.
- Trek geen conclusie over uitvoering, identiteit of bedoeling uit één enkele vermelding.

Onthoud
Artefacten leveren aanwijzingen; conclusies vragen om samenhang en context.

8 Korte samenvatting

Windows-artefacten bieden verschillende vensters op systeem- en bestandsactiviteit. Prefetch, Shimcache, Amcache, MFT, registerhives en Event Logs zijn complementaire bronnen, geen afzonderlijk sluitend bewijs.

- Leg bronnen naast elkaar en interpreteer tijdstempels zorgvuldig.
- Beschrijf wat een artefact ondersteunt én wat het niet zelfstandig kan vaststellen.

Mindmap 098: Hostforensica — Linux- en Unix-artefacten

1. Wat is het?

Systematisch digitale sporen op een Linux- of Unix-host verzamelen en interpreteren, met behoud van context en integriteit.

- Breng gebeurtenissen en mogelijke tijdlijnen in kaart.
- Leg vast wat je waarneemt; scheid waarneming van interpretatie.
- Behandel artefacten als aanwijzingen, niet als zelfstandig bewijs van intentie of daderschap.

2. Welke sporen?

Verschillende artefacten kunnen elkaar aanvullen, maar zijn niet altijd volledig of beschikbaar.

- Systeemlogboeken: gebeurtenissen en services.
- Authenticatielogboeken: onder meer aanmeldpogingen.
- wtmp en utmp: gegevens over aanmeldingen, starts en sessies.
- Shellgeschiedenis: mogelijk uitgevoerde opdrachten, met beperkingen.
- Cron: schema's voor geplande taken.

3. Systeemlogboeken

Syslog is een veelgebruikte loggingvoorziening. Locaties en inhoud hangen af van distributie en configuratie.

- Op Debian en Ubuntu staat syslog vaak in /var/log/syslog.
- Controleer welk systeem logt en welke bestanden aanwezig zijn.
- Noteer bestandsnaam, tijdstempels en relevante metadata.
- Houd rekening met rotatie, ontbrekende perioden en mogelijke klokafwijkingen.

4. Aanmeldsporen

Authenticatiegebeurtenissen kunnen helpen bij het reconstrueren van toegang tot een host.

- Op Debian en Ubuntu staat authenticatie-informatie vaak in /var/log/auth.log.
- Andere distributies kunnen bijvoorbeeld /var/log/secure gebruiken.
- De vastgelegde details zijn afhankelijk van de loggingconfiguratie.
- Vergelijk relevante gebeurtenissen met andere sporen en tijdstempels.

5. Sessies en opstart

wtmp en utmp kunnen informatie geven over aanmeldingen, systeemstarts en actieve sessies.

- last kan gegevens uit wtmp weergeven; who kan huidige sessies tonen.
- wtmp bevat doorgaans aanmeldingen en systeemstarts; utmp doorgaans huidige sessies.
- Bestanden kunnen ontbreken, geroteerd of gewijzigd zijn.
- Een weergegeven sessie is een spoor om te onderzoeken, geen sluitende conclusie.

6. Bash-geschiedenis

Bash kan opdrachten opslaan in ~/.bash_history, maar deze geschiedenis is geen volledig of betrouwbaar auditlog.

- Regels kunnen ontbreken, vertraagd worden opgeslagen of zijn aangepast.
- Opslaan kan uitgeschakeld zijn; niet iedere shell gebruikt dezelfde instellingen.
- Vergelijk een opdrachtregel waar mogelijk met logboeken en andere artefacten.
- Trek uit één historische opdracht geen conclusie over wie handelde of met welke bedoeling.

7. Cron en correlatie

Systeem- en gebruikerscrontabs tonen geplande taken. Interpreteer opdrachten in hun gebruikers-, tijd- en configuratiecontext.

- Noteer welke taak, gebruiker en planning je aantreft.
- Vergelijk geplande tijden met relevante logboek- en sessiesporen.
- Houd rekening met tijdzone, klokafwijking en logrotatie.
- Documenteer een voorbeeld als: 'taak gepland om 02.00 uur'; onderzoek vervolgens of andere artefacten uitvoering ondersteunen.
- Verander bronmateriaal niet tijdens het verzamelen en documenteren.

8. Korte samenvatting

Hostforensica bouwt een zorgvuldig onderbouwd beeld op uit meerdere sporen en hun beperkingen.

- Identificeer relevante bestanden en metadata.
- Bewaar context en integriteit; leg waarnemingen nauwkeurig vast.
- Vergelijk artefacten en tijdstempels, met aandacht voor configuratie, rotatie en tijdverschillen.
- Scheid wat is waargenomen van wat daaruit wordt afgeleid.

Cloudforensiek & vluchtige gegevens

1. Wat is cloudforensiek?

- Digitaal onderzoek naar gegevens uit cloudomgevingen.
- Verzamel en bewaar relevante gegevens zorgvuldig; analyseer ze met aandacht voor herkomst en integriteit.
- Documenteer wat is vastgelegd, wanneer, waar en op welke manier.

2. Cloudmodellen en verantwoordelijkheden

- Gedeelde verantwoordelijkheid: klant en cloudprovider beheren elk bepaalde onderdelen.
- IaaS, PaaS en SaaS verdelen beheer en controle verschillend.
- Breng per dienst in kaart welke onderdelen en gegevens de klant zelf kan beheren.

3. Bronnen in AWS, Azure en GCP

- AWS, Microsoft Azure en Google Cloud Platform hebben verschillende diensten en logbronnen.
- Mogelijke bronnen: account- en identiteitsactiviteit, platformlogs en applicatielogs.
- Voorbeeld: vergelijk een relevante aanmelding met de bijbehorende applicatiegebeurtenis, als beide beschikbaar zijn.

4. Vluchtige gegevens en virtuele machines

- Vluchtige gegevens kunnen verdwijnen wanneer een instantie stopt of opnieuw wordt gestart.
- Voorbeelden: actieve processen en netwerkverbindingen.
- Leg relevante vluchtige gegevens waar mogelijk tijdig vast volgens het goedgekeurde incidentresponsproces; noteer tijdstip, bron en methode.
- Let op: Het veiligstellen van vluchtige gegevens kan de toestand van een systeem beïnvloeden. Volg de afgesproken procedure en leg handelingen vast.

5. Snapshots en opslag

- Een snapshot legt op een bepaald moment de toestand van een ondersteunde schijf of instantie vast.
- Een snapshot kan runtimegegevens, zoals geheugeninhoud, missen.
- Gebruik snapshots niet als vervanging voor logs of een zorgvuldig bewaringsproces.

6. Audittrails en CloudTrail

- Audittrails registreren gebeurtenissen en handelingen.
- AWS CloudTrail registreert AWS-accountactiviteit en API-aanroepen.
- Beschikbare gegevens hangen onder meer af van toegangsrechten, bewaartermijnen en export- en opslaginstellingen.

7. Serverless logs en onderzoekssporen

- Bij serverless-diensten beheert de provider een groot deel van de onderliggende infrastructuur.
- Onderzoek kan steunen op beschikbare platform-, applicatie- en identiteitslogs.
- Logbronnen en dekking verschillen per dienst; logging moet mogelijk vooraf zijn ingeschakeld of ingericht.

8. Korte samenvatting en kernbegrippen

- Cloudforensiek: gegevens zorgvuldig verzamelen, bewaren en analyseren.
- Herkomst, integriteit en documentatie helpen gegevens en handelingen te duiden.
- Controleer per clouddienst welke logs en gegevens beschikbaar zijn.
- Snapshot ≠ volledig beeld; vluchtige gegevens kunnen ontbreken.

Dreigingsmodellering: STRIDE & PASTA

1 Wat is dreigingsmodellering?

Een gestructureerde manier om een systeem, waardevolle onderdelen en mogelijke dreigingen te onderzoeken en passende maatregelen te kiezen.

- Begin bij het systeemdoel en de context.
- Herhaal de analyse als architectuur of risico's veranderen.

Doel
Begrijp waar risico's kunnen ontstaan en leg bevindingen en maatregelen vast.

2. Doel, scope en assets

Bepaal wat je onderzoekt en welke onderdelen bescherming verdienen.

- **Doel**: welk beveiligingsvraagstuk onderzoek je?
- **Scope**: welke applicaties, zones en gegevens vallen binnen de analyse?
- **Assets**: bijvoorbeeld accounts, persoonsgegevens, bestelgegevens en beschikbaarheid.
- Noteer aannames, afhankelijkheden en wat buiten scope blijft.

3. Systeem en dataflowdiagram

Breng componenten en gegevensstromen in kaart met een dataflowdiagram (DFD).

- Toon processen, gegevensopslag, stromen en externe entiteiten.
- Markeer vertrouwensgrenzen tussen zones of partijen.
- Houd diagram en scope begrijpelijk en actueel.

Voorbeeld webwinkel
Klant → browser → webapp → database; de webapp wisselt ook gegevens uit met een betaalprovider. Markeer waar gegevens een vertrouwensgrens passeren.

4. STRIDE-categorieën

Gebruik STRIDE om mogelijke dreigingen systematisch te ordenen. Koppel elke categorie aan concrete onderdelen en gegevensstromen.

- **S — Spoofing**: identiteitsvervalsing, bijvoorbeeld misbruik van een klantaccount.
- **T — Tampering**: manipulatie van gegevens, bijvoorbeeld een gewijzigd bestelbedrag.
- **R — Repudiation**: betwistbaarheid van handelingen; onderzoek of acties aantoonbaar zijn vastgelegd.
- **I — Information disclosure**: ongewenste openbaarmaking, bijvoorbeeld persoonsgegevens.
- **D — Denial of service**: dienstweigering of verlies van beschikbaarheid.
- **E — Elevation of privilege**: ongeoorloofde privilegeverhoging.

Gericht toepassen
Niet elke categorie is op elk systeemonderdeel even relevant. Onderzoek plausibele scenario's per component en stroom.

5. PASTA in zeven stappen

PASTA is een risicogestuurde, iteratieve analysemethode. De stappen helpen de analyse structureren; ze garanderen geen volledige vondst van alle dreigingen.

1. **Doelen definiëren** — Leg bedrijfsdoel en beveiligingsdoelen vast.
2. **Technische scope bepalen** — Bepaal systemen, componenten, grenzen en afhankelijkheden.
3. **Applicatie ontleden** — Beschrijf architectuur, gegevens en gegevensstromen.
4. **Dreigingsanalyse uitvoeren** — Breng relevante dreigingsactoren en scenario's in beeld.
5. **Kwetsbaarheden en zwaktes analyseren** — Onderzoek waar bestaande ontwerpen of maatregelen tekort kunnen schieten.
6. **Aanvallen modelleren** — Werk plausibele aanvalspaden op hoofdlijnen uit voor beveiligingsanalyse.
7. **Risico's analyseren** — Beoordeel risico's, prioriteiten en passende maatregelen.

6. Dreigingsscenario's en voorbeelden

Formuleer een scenario als een mogelijke gebeurtenis met een systeemonderdeel, een dreiging en een gevolg.

- **Klantaccount**: identiteitsvervalsing kan ongeoorloofde toegang tot accountgegevens mogelijk maken.
- **Bestelstroom**: manipulatie van bestelgegevens kan de integriteit van een bestelling aantasten.
- **Database**: onvoldoende bescherming kan persoonsgegevens blootstellen.
- **Webapp**: een beschikbaarheidsprobleem kan bestellen tijdelijk verhinderen.
- Leg vast welk onderdeel en welke gegevensstroom geraakt worden.

Werk zorgvuldig
Onderzoek alleen systemen waarvoor je bevoegd bent. Beschrijf risico's voor verdediging en documentatie, niet als uitvoerbare aanvalsinstructies.

7. Risico-evaluatie en mitigaties

Vergelijk risico's kwalitatief en koppel bevindingen aan proportionele maatregelen en verantwoordelijken.

- Weeg kans, impact, blootstelling en bestaande maatregelen mee.
- Noteer aannames en onzekerheden.
- Prioriteer risicobeperking; vermijd schijnprecisie zonder onderbouwde methode.
- Leg per bevinding de maatregel en eigenaar vast.

Voorbeeld
Als accountmisbruik plausibel is, onderzoek dan passende maatregelen zoals sterke authenticatie, beperkte toegangsrechten en controleerbare logregistratie.

8. Korte samenvatting

Dreigingsmodellering verbindt systeemcontext, assets, dreigingsscenario's, risico's en maatregelen.

- Teken componenten, gegevensstromen en vertrouwensgrenzen.
- Gebruik STRIDE om dreigingen te ordenen; gebruik PASTA als iteratief, risicogestuurd analysekader.
- Beoordeel risico's met expliciete aannames en leg maatregelen en eigenaren vast.
- Werk de analyse bij wanneer systeem of risico's veranderen.

MITRE D3FEND — Tegenmaatregelen & Defensieve Architectuur

1. Doel & Context

MITRE D3FEND is een kennisgraph en ontologie om defensieve technieken en tegenmaatregelen te beschrijven en te ordenen. Het kan naast MITRE ATT&CK worden gebruikt om defensieve keuzes te verkennen.

- Gebruik het als gestructureerde referentie voor defensieve planning.
- Een beschreven tegenmaatregel is geen garantie dat een aanval wordt voorkomen.

2. Kernconcepten

De mindmap groepeert vijf goedgekeurde thema's voor defensieve analyse:

- **Model** — breng systemen, gegevens en relaties in kaart.
- **Deceptie** — gebruik gecontroleerde lokmiddelen, zoals een decoy-resource, om signalen te verzamelen.
- **Isolatie** — beperk verbindingen tussen systemen of netwerkzones.
- **Detectie** — verzamel en beoordeel signalen die op afwijkend gedrag kunnen wijzen.
- **Verificatie** — controleer of een maatregel werkt zoals bedoeld.

3. Relatie met ATT&CK-tactieken

MITRE ATT&CK beschrijft gedrag van tegenstanders; D3FEND beschrijft defensieve technieken. Koppelingen tussen de kaders kunnen helpen om gedrag en mogelijke tegenmaatregelen samen te analyseren.

- Gebruik een koppeling als hulpmiddel voor analyse, niet als bewijs van volledige dekking of gegarandeerde effectiviteit.

4. Toepassing in architectuurontwerp

Kies maatregelen in samenhang met de bedrijfsmiddelen, relevante dreigingen en operationele behoeften.
Een gelaagde architectuur combineert meerdere controles.

- Netwerksegmentatie kan communicatie tussen zones beperken.
- Monitoring kan verdachte activiteit zichtbaar maken.
- Verificatie kan nagaan of een controle zich gedraagt zoals bedoeld.

Voorbeeld

Een organisatie kan gevoelige systemen in een aparte zone plaatsen, loggegevens monitoren en periodiek controleren of de toegangsregels volgens ontwerp werken.

5. Technische componenten

Voorbeelden van componenten die in een defensieve architectuur kunnen bijdragen:

- Netwerksegmentatie om zones af te bakenen.
- Logging en monitoring om gebeurtenissen te volgen.
- Endpointcontroles voor apparaten en werkstations.
- Toegangscontroles om rechten te begrenzen.
- Decoy-resources als gecontroleerde lokmiddelen.

6. Uitdagingen & Complexiteit

De waarde van een maatregel of koppeling hangt af van de uitvoering en de omgeving. Beoordeel keuzes daarom in hun operationele context.

- Implementatie en configuratie beïnvloeden het gedrag van controles.
- Beschikbare telemetrie bepaalt welke signalen zichtbaar zijn.
- Omgevingen veranderen; onderhoud en periodieke herbeoordeling blijven nodig.
- Een mapping tussen technieken bewijst op zichzelf niet dat een risico is afgedekt.

7. Praktische defensiestrategie

Werk in een herhaalbare cyclus voor legitieme defensieve planning:

1. **Inventariseer bedrijfsmiddelen**
   Breng systemen, gegevens, netwerkzones en hun belang in kaart.
2. **Bepaal relevante risico's**
   Koppel mogelijke dreigingen aan de bedrijfsmiddelen en operationele behoeften.
3. **Kies aanvullende controles**
   Combineer passende maatregelen; vertrouw niet op één controle als volledige oplossing.
4. **Valideer bevoegd en veilig**
   Controleer maatregelen in een geautoriseerde test- of beheeromgeving.
5. **Monitor en verfijn**
   Volg relevante signalen en pas maatregelen aan wanneer systemen of behoeften veranderen.

8. Korte samenvatting

D3FEND is een gestructureerde referentie voor defensieve technieken en planning. In combinatie met ATT&CK kunnen de kaders analyse ondersteunen, maar D3FEND is geen kant-en-klare oplossing en biedt geen garantie op bescherming.

- Breng systemen en risico's in kaart.
- Combineer maatregelen en stem ze af op de omgeving.
- Controleer, monitor en onderhoud de gekozen controles.

Incidentrespons: playbooks & runbooks

1 Playbook of runbook?

Incidentresponsdocumentatie beschrijft hoe een team een incident aanpakt. De termen kunnen per organisatie deels overlappen.

- **Playbook:** aanpak op hoofdlijnen, met doelen, rollen, beslispunten en mogelijke acties voor een incidenttype.
- **Runbook:** doorgaans gedetailleerde, herhaalbare stappen voor een specifieke taak.
- **Voorbeeld:** een playbook beschrijft de aanpak bij een vermoedelijk gecompromitteerd account; een runbook legt een vaste controle- of hersteltaak stap voor stap uit.

Onthoud
Leg vast hoe jouw organisatie de termen gebruikt; er is geen universele grens.

2 Doel en consistentie

Duidelijke documentatie helpt het team rollen, communicatie, escalatie en beslissingen consistent te organiseren.

- Wijs rollen en bevoegdheden toe.
- Beschrijf wie wordt geïnformeerd en wanneer.
- Maak beslispunten en vervolgstappen vindbaar.
- Stem procedures af op de eigen systemen en werkwijze.

3 Mogelijke procesfasen

Een incidentresponsproces kan uit verschillende fasen bestaan. De precieze indeling verschilt per raamwerk en organisatie.

- Voorbereiding.
- Detectie en analyse.
- Indamming.
- Uitroeiing.
- Herstel.
- Evaluatie en geleerde lessen.

Gebruik als leidraad
Pas de fasen aan de context aan; dit is geen universele verplichte volgorde.

4 Automatisering met SOAR

SOAR kan terugkerende stappen orkestreren en automatiseren, zodat informatie en taken samenkomen in een workflow.

- **Voorbeeld:** verrijk een waarschuwing met beschikbare context.
- **Voorbeeld:** maak automatisch een zaak aan voor opvolging.
- Controleer autorisaties en laat risicovolle acties passend beoordelen.
- Stem automatisering af op de eigen omgeving.

Veilig automatiseren
Automatisering vervangt geen passende autorisatie of menselijke beoordeling van risicovolle acties.

5 Versiebeheer en tabletop-oefeningen

Beheer wijzigingen traceerbaar en bespreek procedures vooraf met de mensen die ze moeten toepassen.

- Noteer wijzigingen en houd versies bij.
- Gebruik tabletop-oefeningen: bespreek een gesimuleerd scenario en toets rollen, beslissingen en communicatie.
- Leg gevonden lacunes vast en werk procedures waar nodig bij.
- Een tabletop is een besprekingsoefening, geen live incidentrespons.

6 Valkuilen en veroudering

Een procedure kan onbruikbaar worden als de omgeving verandert maar de documentatie achterblijft.

- Controleer contactpersonen, systemen, bevoegdheden en afhankelijkheden.
- Plan periodieke controle en neem wijzigingen mee in het beheer.
- Gebruik oefeningen om onduidelijke of verouderde stappen te ontdekken.
- Maak geen stappen automatisch geldig voor elke organisatie.

Let op
Een verouderd contact, systemndetail of bevoegdheid kan de uitvoering vertragen.

7 Effectiviteit meten: MTTD en MTTR

Deze gemiddelden kunnen inzicht geven in onderdelen van incidentrespons, maar zeggen op zichzelf niet alles over de kwaliteit ervan.

- **MTTD:** doorgaans gemiddelde tijd tot detectie.
- **MTTR:** doorgaans gemiddelde tijd tot herstel of respons.
- Leg voor beide vast wanneer de meting begint en eindigt.
- Gebruik consistente definities en vergelijk meetperioden zorgvuldig.
- Trek geen conclusie over kwaliteit op basis van één cijfer.

Meetdefinities
De precieze start- en eindpunten kunnen verschillen; documenteer ze voordat je resultaten vergelijkt.

8 Korte samenvatting

Goede incidentresponsdocumentatie maakt afspraken duidelijk, uitvoerbaar en aanpasbaar.

- Een playbook beschrijft doorgaans de aanpak; een runbook de uitvoeringsstappen.
- Gebruik passende rollen, beslispunten en mogelijke procesfasen.
- Automatiseer terugkerend werk zorgvuldig en beheer wijzigingen traceerbaar.
- Oefen, actualiseer documentatie en definieer meetpunten helder.

Geheugenbeschermings-mechanismen

1. Grondbeginsel van geheugenaanvallen

Geheugenaanvallen maken misbruik van programmeerfouten om gegevens te overschrijven of de uitvoering van een programma te beïnvloeden.

- Geheugencorruptie is een algemene risicocategorie, geen afzonderlijke aanvaltechniek.
- Veilig programmeren en zorgvuldig testen helpen fouten te voorkomen.

2. Data Execution Prevention (DEP / NX-bit)

DEP markeert bepaalde geheugen-pagina's als niet-uitvoerbaar. De NX-bit kan besturingssystemen helpen die beperking af te dwingen.

- Bemoeilijkt uitvoering van code uit geheugen dat voor gegevens is bedoeld.
- Voorkomt niet elk misbruik van geheugencorruptie.

3. Address Space Layout Randomization (ASLR)

ASLR varieert bij het laden adressen van onderdelen zoals programma's, bibliotheken en stapelgeheugen.

- Maakt geheugenadressen minder voorspelbaar.
- De effectiviteit hangt onder meer af van ondersteuning en configuratie.

4. Stack Canaries & SafeSEH

Een stack canary is een controlewaarde op de stapel. Een afwijking kan worden ontdekt voordat een functie terugkeert. SafeSEH registreert en controleert geldige exception handlers in Windows 32-bits uitvoerbare bestanden.

- Canaries helpen bepaalde overschrijvingen op te merken, maar voorkomen niet elke vorm van geheugencorruptie.
- SafeSEH heeft een beperkte context; het is geen algemene bescherming voor alle Windows-versies of 64-bits processen.

5. Control Flow Integrity (CFI)

CFI is een familie maatregelen die grenzen stelt aan toegestane indirecte sprongen en functieaanroepen.

- Kan afwijkingen van een vooraf bepaald controlepad tegenhouden of detecteren.
- De precieze werking verschilt per implementatie.

6. Bypass-concepten: ROP/JOP

ROP en JOP zijn conceptuele voorbeelden van het hergebruiken van bestaande instructiesequenties om de uitvoering om te leiden.

- ROP gebruikt doorgaans korte sequenties die eindigen met een return-instructie.
- JOP gebruikt indirecte sprongen.
- Dit overzicht bevat geen uitvoeringsstappen of code.

Onthoud Deze concepten laten zien waarom één enkele geheugenmaatregel niet alle risico's wegneemt.

7. Implementatie op Windows/Linux

Windows en Linux kunnen ondersteuning bieden voor onder meer niet-uitvoerbaar geheugen, adresrandomisatie en compiler-gebaseerde controles.

- Beschikbaarheid en werking hangen af van besturingssysteem, architectuur, compiler en toepassing.
- Ga niet ervan uit dat een mechanisme overal standaard is ingeschakeld.

8. Korte samenvatting

DEP, ASLR, canaries, SafeSEH en CFI beperken verschillende aspecten van geheugenrisico's. Ze zijn niet uitwisselbaar en geen ervan biedt op zichzelf volledige bescherming.

- Gebruik gelagde mitigaties waar passend.
- Blijf veilig programmeren, systemen bijwerken en configuraties zorgvuldig beheren.

Rootkits en bootkits

1. Definitie

- Een rootkit is malware of een verzameling technieken die activiteit kan verbergen en verhoogde toegang kan ondersteunen.
- User-mode: werkt in de gebruikersruimte. Kernel-mode: werkt met kernelprivileges. Firmware-rootkits richten zich op firmware, die vóór of onafhankelijk van het besturingssysteem kan draaien.

2. Hooking en DKOM

- Hooking onderschept of wijzigt functieaanroepen of andere uitvoeringspunten; dit kan informatie verbergen of gedrag beinvloeden.
- DKOM (Direct Kernel Object Manipulation) wijzigt kernelgegevensstructuren, bijvoorbeeld om een proces uit een gebruikelijke proceslijst te verbergen.
- Vergelijking van verschillende systeemweergaven kan inconsistenties aan het licht brengen. Hooking kan ook legitiem worden gebruikt.

3. Bootkits: UEFI en MBR

- Een bootkit richt zich op het opstartproces om vóór of tijdens het laden van het besturingssysteem persistentie te verkrijgen.
- Historische varianten konden de MBR of bootloader aanpassen. UEFI-gerichte varianten kunnen firmware of EFI-systeempartities misbruiken.
- MBR en UEFI zijn verschillende opstartarchitecturen.

4. Secure Boot en TPM

- Secure Boot controleert tijdens het opstarten digitale handtekeningen van bootcomponenten aan de hand van vertrouwde sleutels en beleid.
- Een TPM kan sleutels beschermen en opstartmetingen ondersteunen. Attestatie kan helpen beoordelen of de gemeten opstartstatus aan verwacht beleid voldoet.
- Deze barrières verkleinen risico's, maar voorkomen of verwijderen niet zelfstandig alle malware.

5. Detectie

- Gebruik waar passend geheugenscans vanuit een offline of vertrouwde omgeving.
- Vergelijk actieve systeemweergaven en controleer boot- en firmwareconfiguratie.
- Vergelijk bestanden of metingen met bekende goede referenties. Geen enkele methode is op zichzelf afdoende.

6. Verwijderingscomplexiteit

- Verborgen componenten kunnen scans ontwijken; persistentie kan op meerdere systeemlagen zitten.
- Een verdedigende aanpak is isoleren, bewijs zorgvuldig bewaren en herstel via vertrouwde middelen of herinstallatie overwegen.
- Firmwareherstel kan specialistische procedures vereisen; herstel is niet gegarandeerd.

7. Forensische sporen

- Mogelijke aanwijzingen: afwijkende geheugenstructuren of hooks en verschillen tussen systeemlijsten.
- Controleer ook onverwachte wijzigingen aan bootconfiguratie of EFI-bestanden, statuswijzigingen van firmware of Secure Boot, en relevante log- of tijdlijnsporen.
- Eén aanwijzing afzonderlijk bewijst geen infectie.

8. Korte samenvatting

- Rootkits kunnen zich op verschillende systeemlagen verbergen; bootkits richten zich op het opstartproces.
- Secure Boot, TPM-metingen en detectiemethoden helpen risico's te beperken en afwijkingen te onderzoeken, maar bieden geen absolute garantie.
- Beoordeel aanwijzingen in samenhang en kies bij vermoedelijke besmetting voor zorgvuldig, vertrouwd herstel.

Veilige netwerkarchitectuur: ZTNA

Zero Trust Network Access (ZTNA) richt toegang op specifieke applicaties en bronnen.

Gebruik de acht modules als compacte begrippenkaart.

1. Wat is ZTNA

- Zero Trust Network Access: toegang op basis van beleid.
- Geef geautoriseerde gebruikers toegang tot specifieke applicaties of bronnen.
- Toegang tot één applicatie betekent niet automatisch toegang tot het hele interne netwerk.

2. Minste privilege op netwerkniveau

- Geef alleen de netwerktoegang die nodig is voor de taak.
- Beperk rechten en reikwijdte zo veel mogelijk.
- Voorbeeld: toegang tot één werkapp, niet tot alle interne systemen.

3. Contextbewuste toegang

- Beleid kan rekening houden met identiteit, apparaatstatus en locatie of netwerkcontext.
- Welke signalen meetellen, hangt af van de inrichting.
- Voorbeeld: toegang kan anders worden beoordeeld op een beheerd apparaat dan op een onbekend apparaat.

4. SDP-architectuur

SDP staat voor Software-Defined Perimeter.

- Een architectuurbenadering waarin beleid bepaalt welke diensten zichtbaar en bereikbaar zijn.
- Toegang wordt verleend aan een geautoriseerde gebruiker of een geautoriseerd apparaat.
- De concrete inrichting kan per organisatie en product verschillen.

5. ZTNA versus traditionele VPN

- Een VPN voor externe toegang maakt doorgaans een versleutelde verbinding met een netwerk of netwerksegment.
- ZTNA richt toegang doorgaans op afzonderlijke applicaties of bronnen.
- Mogelijkheden verschillen per product en architectuur; geen van beide garandeert op zichzelf veiligheid.

Vergelijk zorgvuldig

Kijk naar het toegangsmodel en de inrichting; de naam van een oplossing zegt niet alles over de beveiliging.

6. Integratie met identiteitsproviders

Een identity provider (IdP), oftewel identiteitsprovider, kan gebruikers authenticeren en identiteitsinformatie aan toegangsbeleid leveren.

- Authenticatie bevestigt de identiteit.
- Autorisatie bepaalt welke applicaties of bronnen toegankelijk zijn.
- Combineer identiteitsbeleid zorgvuldig met apparaat- en toegangsregels.

7. Risico's en implementatiefasen

Aandachtspunten zijn onder meer onjuiste beleidsregels, onbekende applicaties of afhankelijkheden, zwakke identiteits- of apparaatcontroles en verstoring door een onvolledige migratie.

- Een algemeen voorbeeld van gefaseerde invoering: inventariseer applicaties en afhankelijkheden.
- Start daarna met een beperkte pilot.
- Pas beleid aan op basis van bevindingen en breid vervolgens uit met monitoring.
- Dit is een voorbeeld, geen universeel verplicht stappenplan.

Let op

Onvolledig inzicht in applicaties en afhankelijkheden kan leiden tot toegangsproblemen of verstoring.

8. Korte samenvatting

- ZTNA verschuift de nadruk van brede netwerktoegang naar expliciete toegang tot specifieke bronnen.
- Identiteit, apparaatbeleid, context en minimale rechten kunnen meespelen in de beslissing.
- Zorgvuldig beleid, monitoring en beheer blijven belangrijk.

SASE & SSE

1. Definitie en Samenstelling

SASE (Secure Access Service Edge) brengt netwerkverbindingen en beveiligingsdiensten samen als cloudgebaseerde dienstverlening. SSE (Security Service Edge) omvat de beveiligingsdiensten binnen dit model.

- SSE omvat onder meer CASB, SWG en ZTNA.
- Het netwerkdeel, vaak SD-WAN, kan met de beveiligingsdiensten worden geïntegreerd.

2. SD-WAN-integratie

SD-WAN (Software-Defined Wide Area Network) beheert software-gestuurd netwerkverkeer tussen locaties, cloudomgevingen en gebruikers.

- Binnen SASE kan verkeer naar cloudbeveiligings-diensten worden geleid.
- SD-WAN en SASE zijn verwant, maar betekenen niet hetzelfde.

3. CASB

CASB (Cloud Access Security Broker) helpt beveiligingsbeleid toe te passen op het gebruik van clouddiensten.

- Geeft inzicht in welke clouddiensten worden gebruikt.
- Kan controles toepassen op toegang en het delen van gegevens.

4. Secure Web Gateway (SWG)

SWG (Secure Web Gateway) controleert webverkeer en kan beleid toepassen op toegang tot websites.

- Kan toegang tot onveilige of niet-toegestane websites beperken.
- Werkt aanvullend op de andere beveiligingsdiensten binnen SSE.

5. ZTNA-integratie

ZTNA (Zero Trust Network Access) verleent toegang tot specifieke applicaties op basis van identiteit en context.

- Geeft niet automatisch brede toegang tot een netwerk.
- Authenticatie en toegangsbeleid blijven relevant tijdens de toegang.

6. Voordelen voor hybride organisaties

Een geïntegreerde cloudarchitectuur kan het beheer van beleid voor gebruikers op verschillende locaties vereenvoudigen.

- Netwerk- en beveiligingsdiensten kunnen dichter bij gebruikers en clouddiensten beschikbaar zijn.
- Mogelijke voordelen hangen af van de inrichting; resultaten zijn niet gegarandeerd.

7. Implementatie-uitdagingen

De invoering vraagt afstemming tussen bestaande systemen, netwerkverbindingen en beveiligingsbeleid.

- Integratie en migratie vanuit bestaande systemen.
- Beleidsafstemming en beheer van identiteit en toegangsregels.
- Afhankelijkheid van de gekozen dienstverlener en aandacht voor netwerkprestaties.

8. Korte samenvatting

SASE combineert netwerk- en beveiligingsfuncties. SSE staat voor de beveiligingsdiensten binnen die architectuur.

- SD-WAN verzorgt softwaregestuurd netwerk-verkeer; CASB helpt beleid voor clouddiensten toe te passen.
- SWG controleert webverkeer; ZTNA geeft toepassingsgerichte toegang op basis van identiteit en context.
- De functies vullen elkaar aan, maar zijn niet onderling uitwisselbaar.

CNAPP: geïntegreerde cloudbeveiliging

1. Definitie van CNAPP

CNAPP (Cloud-Native Application Protection Platform) is een geïntegreerde aanpak of platform voor de beveiliging van cloudtoepassingen en hun cloudinfrastructuur, van ontwikkeling tot uitvoering.

- Brengt beveiligingsmogelijkheden samen die anders over losse hulpmiddelen verdeeld kunnen zijn.
- De mogelijkheden verschillen per product.

2. Convergentie van CSPM en CWPP

Twee belangrijke soorten mogelijkheden die binnen een CNAPP kunnen samenkomen:

- CSPM (Cloud Security Posture Management): helpt risico's in cloudconfiguraties vinden, zoals openbaar toegankelijke opslag.
- CWPP (Cloud Workload Protection Platform): beschermt workloads, zoals virtuele machines en containers, ook tijdens uitvoering.

3. CIEM

CIEM (Cloud Infrastructure Entitlement Management) helpt overmatige toegangsrechten en machtigingsrisico's in cloudomgevingen herkennen en beheren.

- Richt machtigingen zo beperkt mogelijk in: geef alleen toegang die nodig is voor de taak.
- Voorbeeld: controleer of een account meer cloudrechten heeft dan het werk vereist.

4. CI/CD-beveiliging (shift-left)

Shift-left betekent dat beveiligingscontroles eerder in het ontwikkelproces plaatsvinden, vóór implementatie.

- Controleer infrastructuur-als-code en afhankelijkheden.
- Voer beveiligingstests uit in de CI/CD-pijplijn.
- Beoordeel bevindingen en pak ze aan voordat een wijziging wordt geïmplementeerd.

5. Runtimebewaking van containers

Runtimebewaking observeert actieve containerworkloads op beveiligingsrelevant gedrag, configuratie en mogelijke dreigingen.

- Helpt signalen tijdens uitvoering zichtbaar te maken.
- Vult controles tijdens bouwen en configureren aan; vervangt ze niet.

6. Prioritering van kwetsbaarheden

Een ernstscore alleen bepaalt niet altijd welke kwetsbaarheid als eerste aandacht verdient.

- Weeg onder meer uitbuitbaarheid, blootstelling en belang van het betrokken onderdeel mee.
- Betrek beschikbare herstelopties en context.
- Er is geen universele rangschikkingsformule voor elke omgeving.

7. Compliance en governance

Mogelijkheden voor compliance en governance kunnen cloudconfiguraties en beveiligingsmaatregelen koppelen aan geselecteerde organisatie-eisen en inzicht geven in hiaten.

- Ondersteunt zichtbaarheid en opvolging van bevindingen.
- Een platform garandeert op zichzelf geen naleving.

8. Korte samenvatting

CNAPP brengt cloudbeveiligingsmogelijkheden samen over ontwikkeling en uitvoering. Gebruik de onderdelen als aanvulling op elkaar: controleer configuraties, workloads, toegangsrechten, wijzigingen, actieve containers, kwetsbaarheden en geselecteerde eisen.

- Stem prioriteiten af op de context van de cloudomgeving.
- De precieze mogelijkheden verschillen per platform.

Beveiliging van infrastructuur als code (IaC)

1. Wat is IaC?

- Infrastructuur beschrijven en beheren met machineleesbare configuraties.
- Automatiseert het aanmaken en aanpassen van infrastructuur.
- Voorbeelden: Terraform, Ansible en AWS CloudFormation.

2. Veelvoorkomende misconfiguraties

- Een opslaglocatie onbedoeld openbaar maken.
- Toegangsrechten ruimer instellen dan nodig.
- Opslag niet versleutelen.
- Onvoldoende logging inschakelen.

3. Statische IaC-scanners

- Controleren configuratiebestanden op bekende beveiligingsrisico's vóór uitrol.
- Voorbeelden: Checkov en tfsec.
- Onderzoek bevindingen en pas de configuratie aan voordat wijzigingen doorgaan.

4. Beleid als code

- Leg beveiligingsregels vast in code.
- Toets configuraties automatisch aan die regels.
- OPA (Open Policy Agent) is een beleidsengine die zulke controles kan uitvoeren.

5. Geheimenbeheer in sjablonen

- Zet geen wachtwoorden, tokens of sleutels hardgecodeerd in sjablonen.
- Bewaar geheimen in een geschikte geheimenbeheeroplossing.
- Beperk toegang tot geheimen tot wie ze nodig heeft.

Let op
Een geheim in een sjabloon kan meekomen in versiebeheer of andere systemem.

6. Drift-detectie in productie

- Drift is het verschil tussen de gewenste configuratie en de feitelijke infrastructuur.
- Vergelijk beide toestanden om afwijkingen te signaleren.
- Onderzoek afwijkingen en bepaal of de configuratie of de infrastructuur moet worden aangepast.

7. CI/CD-validatiestappen

- Controleer wijzigingen in de configuratie.
- Voer statische beveiligingsscans uit.
- Toets de configuratie aan beleidsregels.
- Laat goedkeuring en uitrol pas volgen nadat bevindingen zijn beoordeeld.
- Controleer na uitrol opnieuw op afwijkingen.

8. Korte samenvatting

- Schrijf infrastructuur als machineleesbare configuratie.
- Voorkom onveilige instellingen en hardgecodeerde geheimen.
- Scan en toets beleid vóór uitrol; controleer na uitrol op drift.

110. Server-Side Template Injection (SSTI)

1. Wat is template-injectie?

- SSTI ontstaat wanneer invoer van een gebruiker als sjabloonsyntaxis wordt verwerkt in plaats van als gewone gegevens.
- De server kan daardoor een sjabloonexpressie evalueren op een plek waar alleen tekst was bedoeld.
- Het kernprobleem: gegevens en uitvoerbare sjabloncode zijn niet goed gescheiden.

2. Sjabloonengines

- Jinja2, Twig en FreeMarker zijn voorbeelden van sjabloonengines.
- Elke engine heeft eigen syntaxis en mogelijkheden; gedrag hangt ook af van configuratie en context.
- Een tekenreeks die in de ene engine betekenis heeft, hoeft dat niet te hebben in een andere.

3. Detectie en fuzzing

- Test alleen in een eigen of uitdrukkelijk toegestane omgeving.
- Gebruik onschadelijke invoer en observeer of de server die als sjabloonexpressie behandelt.
- Fuzzing test gecontroleerde varianten; een afwijkende respons is een aanwijzing, geen bewijs van exploiteerbaarheid.
- Leg vast: invoer, context, engine voor zover bekend en waargenomen uitvoer.

4. Van expressie tot RCE

- Een eerste gevolg kan onverwachte evaluatie of gewijzigde uitvoer zijn.
- Ernstigere gevolgen hangen af van beschikbare objecten, enginefuncties, privileges en isolatie.
- Remote Code Execution (RCE) is in sommige configuraties een mogelijk risico, niet een onvermijdelijk gevolg van iedere SSTI.

5. Kwetsbare contexten in webapps

- Sjablonen die dynamisch worden opgebouwd met niet-vertrouwde invoer.
- Gebruikersinvoer die wordt samengevoegd met sjabloonbroncode.
- Functies waarmee gebruikers eigen sjablonen kunnen aanpassen.
- Let op de grens tussen invoer, sjabloonbroncode en gerenderde uitvoer.

6. Verdediging en contextuele outputcodering

- Behandel niet-vertrouwde invoer als gegevens; voeg die niet samen met sjabloonbroncode.
- Gebruik vaste sjablonen en geef gegevens via daarvoor bedoelde variabelen door.
- Pas outputcodering toe op de uitvoercontext, bijvoorbeeld HTML-tekst of een HTML-attribuut.
- Outputcodering is aanvullend en vervangt geen veilige sjabloonconstructie.

7. Sandbox: beperkingen en realiteit

- Een sandbox kan toegankelijke objecten en mogelijkheden beperken.
- Bescherming hangt af van engine, configuratie, toegestane functies en omgeving.
- Een sandbox garandeert niet dat een kwetsbaar ontwerp veilig is; herstel de oorzaak.

8. Korte samenvatting

- Scheid gegevens van sjabloncode.
- Herken dat gedrag engine- en contextafhankelijk is.
- Test uitsluitend gecontroleerd; zie testresultaten als aanwijzingen.
- Herstel onveilige sjabloonconstructie en vertrouw niet alleen op sandboxing.

Webapplicatieaanval: XML External Entity (XXE)-injectie

1. XML en DTD-entiteiten

XML ordent gegevens met elementen en attributen. Een DTD (Document Type Definition) kan de documentstructuur en entiteiten declareren.

- Een XML-entiteit is een benoemde verwijzing die de parser door tekst kan vervangen.
- Een externe entiteit verwijst naar gegevens buiten het XML-document, bijvoorbeeld via een URI.
- Wat de parser met entiteiten doet, hangt af van de configuratie.

2. XXE uitgelegd

XXE (XML External Entity) ontstaat wanneer een applicatie niet-vertrouwde XML verwerkt en de parser externe entiteiten onveilig afhandelt.

- De parser kan een externe entiteit proberen op te halen en te verwerken.
- Daardoor kan een aanvaller proberen gegevens of netwerkverzoeken via de server te laten lopen.

3. Gevolgen: bestandslezing, SSRF en DoS

Onveilige verwerking van externe entiteiten kan gevolgen hebben voor vertrouwelijkheid, netwerktoegang en beschikbaarheid.

- **Bestandslezing:** de server kan worden aangezet tot het benaderen van bestanden waartoe het proces toegang heeft.
- **SSRF:** de server kan worden aangezet tot een netwerkverzoek naar een andere bestemming.
- **DoS:** geneste entiteiten kunnen zeer veel tekst opleveren en zo geheugen of verwerkingstijd uitputten.
- De Billion Laughs-aanval is een bekend conceptueel voorbeeld van sterke entiteitsuitbreiding.

4. Blinde XXE en OOB

Bij blinde XXE verschijnen uitgelezen gegevens niet rechtstreeks in het antwoord. Een externe interactie kan toch aanwijzingen geven over de verwerking.

- OOB (out-of-band) verwijst naar communicatie via een apart kanaal.
- Bij een geautoriseerde test kan een tester gecontroleerde waarnemingen gebruiken om zulke interacties vast te stellen.
- Gebruik alleen onschadelijke, gecontroleerde invoer in een omgeving waarvoor je toestemming hebt.

Let op
Geen zichtbare uitvoer sluit XXE niet uit; beoordeel ook de verwerking, configuratie en logs.

5. Detectie

Onderzoek waar XML wordt verwerkt en hoe de parser is ingesteld. Voer tests uitsluitend uit binnen een geautoriseerde omgeving.

- Breng XML-verwerkingspaden en gebruikte parserconfiguraties in kaart.
- Controleer logs en relevante beveiligingswaarnemingen op onverwachte externe interacties.
- Gebruik onschadelijke, gecontroleerde testinvoer; vermijd toegang tot echte bestanden of systemen.
- Beoordeel de uitkomst ook wanneer het antwoord geen gegevens toont.

6. Verdediging: DTD uitschakelen

Beperk parsergedrag dat externe gegevens kan ophalen of entiteiten onbeperkt kan uitbreiden.

- Schakel DTD-verwerking uit wanneer die niet nodig is.
- Blokkeer externe entiteiten en het ophalen van externe schema's.
- Stel grenzen in voor documentgrootte, entiteitsuitbreiding en verwerkingstijd.
- Controleer dat de gekozen instellingen daadwerkelijk gelden voor de gebruikte parser.

Praktische vuistregel
Beperk functionaliteit die de toepassing niet nodig heeft en valideer de configuratie in de eigen omgeving.

7. Veilige parserkeuze per programmeertaal

Kies een parser die externe entiteiten standaard uitschakelt of expliciet veilig kan worden geconfigureerd. API's en standaardinstellingen verschillen per bibliotheek en versie.

- **Java:** controleer de documentatie van de gebruikte XML-parser en stel externe-entiteitsverwerking expliciet veilig in.
- **.NET:** controleer de documentatie van de gebruikte XML-parser en de configuratie voor DTD's en externe bronnen.
- **PHP:** controleer de documentatie van de gebruikte XML-bibliotheek en de relevante parserinstellingen.
- **Python:** controleer de documentatie van de gebruikte XML-bibliotheek en de verwerking van entiteiten en externe bronnen.
- **JavaScript-runtimebibliotheken:** controleer de documentatie van de gekozen bibliotheek; instellingen verschillen.
- Vertrouw niet alleen op de taalnaam: beoordeel bibliotheek, versie en actieve configuratie.

Controleer de details
Neem niet aan dat een parser zonder configuratiecontrole veilig is; raadpleeg de documentatie van de gebruikte versie.

8. Korte samenvatting

XXE berust op onveilige parserverwerking van externe XML-entiteiten. Mogelijke gevolgen zijn bestandslezing, SSRF en uitputting van systeembronnen.

- Beperk of schakel DTD- en externe-entiteitsverwerking uit wanneer die niet nodig is.
- Stel grenzen in voor grootte, uitbreiding en verwerkingstijd.
- Controleer parserinstellingen en test veilig en geautoriseerd.

Webapplicatieaanval: onveilige deserialisatie

1. Serialisatie en deserialisatie

Serialisatie zet gegevens of objecten om in een vorm die kan worden opgeslagen of verzonden. **Deserialisatie** reconstrueert daaruit gegevens of objecten.

- Denk aan het inpakken en uitpakken van gegevens.
- Het herstelde resultaat hangt af van het gebruikte formaat en de software.

2. Risico van niet-vertrouwde objectstromen

Een objectstroom is riskant wanneer een toepassing **niet-vertrouwde** geserialiseerde invoer verwerkt zonder passende beperkingen of validatie.

- Invoer kan van een gebruiker, bestand, bericht of externe dienst komen.
- Mogelijke gevolgen hangen af van taal, bibliotheek en toepassing.
- Een onveilige verwerking leidt niet automatisch in elk geval tot code-uitvoering.

3. Gadget chains en magic methods

Een gadget chain is een reeks bestaande programmacomponenten die tijdens objectherstel onbedoeld kunnen samenwerken met een schadelijk effect. Magic methods zijn speciale methoden die een taal of framework automatisch kan aanroepen bij bepaalde gebeurtenissen, zoals objectconstructie of -herstel.

- De details verschillen per taal, framework en bibliotheek.
- De combinatie van beschikbare klassen en automatisch aangeroepen methoden kan relevant zijn voor het risico.

4. Mogelijke impact

Afhankelijk van de omstandigheden kan onveilige deserialisatie leiden tot objectmanipulatie, andere beveiligingsproblemen of uitvoering van ongewenste code.

- Een mogelijk scenario is code-uitvoering op de server.
- Een ander mogelijk scenario is misbruik van de rechten van het proces.
- Dit zijn mogelijke gevolgen, geen onvermijdelijke uitkomst van elke onveilige deserialisatie.

Belangrijk
Beoordeel impact in de context van de toepassing en de rechten waarmee het proces draait.

5. Voorbeelden uit Java, Python en PHP

Voorbeelden van aandachtspunten per taal:

- **Java**: sommige objectdeserialisatiebibliotheken kunnen bij het herstellen van objectgrafen methoden van klassen aanroepen.
- **Python**: pickle is bedoeld voor vertrouwde gegevens; het laden van niet-vertrouwde pickle-inhoud kan code uitvoeren.
- **PHP**: unserialize kan objecten herstellen. Afhankelijk van beschikbare klassen en methoden kunnen onveilige neveneffecten ontstaan.
- Geen van deze voorbeelden betekent dat iedere bibliotheek, configuratie of versie kwetsbaar is.

6. Detectie en analyse

Onderzoek defensief hoe gegevens worden verwerkt. Werk met toestemming en gebruik voor verdachte gegevensstructuren een geïsoleerde omgeving.

- Stel vast welk gegevensformaat en welke bibliotheek worden gebruikt.
- Volg het invoerpad van bron tot verwerking en bepaal waar de vertrouwensgrens ligt.
- Onderzoek relevante gegevensstructuren en logboeken in een geïsoleerde omgeving.
- Een verdachte payload op zichzelf bewijst niet dat code is uitgevoerd.

7. Verdediging: veilige gegevensuitwisseling en typevalidatie

Voor gegevensuitwisseling is een eenvoudig **gegevensformaat zoals JSON doorgaans veiliger** dan het herstellen van willekeurige objecten. JSON is op zichzelf echter geen volledige beveiligingsmaatregel.

- Valideer het schema en de verwachte typen en waarden.
- Beperk de omvang en controleer de context van de invoer.
- Sta waar passend alleen expliciet toegestane typen toe.
- Vermijd automatische objectconstructie op basis van invoer.
- Beperk deserialisatie en geef het toepassingsproces alleen de noodzakelijke rechten.

Praktische vuistregel
Behandel objectstromen als onbetrouwbaar tenzij herkomst en integriteit zijn gewaarborgd.

8. Korte samenvatting

Veilige verwerking begint bij het herkennen van de vertrouwensgrens en het beperken van wat invoer kan doen.

- Serialisatie verpakt gegevens; deserialisatie herstelt ze.
- Onveilige verwerking van niet-vertrouwde objeuitstromen kan beveiligingsrisico's veroorzaken.
- Gebruik passende beperkingen en validatie; vertrouw niet op JSON alleen.
- Beperk rechten en onderzoek verdachte invoer zorgvuldig.

Geavanceerde phishing-infrastructuur: reverse proxy & AiTM

1. Evolutie van phishing

- Phishing misleidt iemand om gegevens te delen of in te loggen op een nagemaakte pagina.
- Moderne varianten kunnen verkeer via een tussenliggende proxy leiden.
- Doel: niet alleen gegevens bemachtigen, maar soms ook een aangemelde sessie misbruiken.

2. Werking van AiTM

- AiTM betekent Adversary-in-the-Middle: een aanvaller plaatst een proxy tussen browser en echte dienst.
- De proxy geeft aanmeldverkeer door tussen beide kanten.
- Afhankelijk van dienst en configuratie kunnen gegevens of een sessiewaarde worden onderschept.
- MFA voorkomt dit niet altijd; phishing-resistente authenticatie biedt sterkere bescherming.

Op hoofdlijnen

De browser denkt met de dienst te communiceren; de tussenliggende proxy geeft verkeer door. Dit is een uitleg van het risico, geen handleiding.

3. Sessie-cookies en tokens

- Na authenticatie kan een sessiecookie of toegangstoken de ingelogde sessie vertegenwoordigen.
- Een geldige onderschepte waarde kan mogelijk toegang geven zonder het volledige aanmeldproces opnieuw te doorlopen.
- Risico hangt onder meer af van de implementatie en geldigheidsduur.

4. Open-source tooling

- Evilginx is een naam die je in beveiligingsonderzoek rond AiTM-achtige phishingframeworks kunt tegenkomen.
- Herken de context en bespreek hulpmiddelen uitsluitend op hoofdlijnen.
- Installatie-, configuratie- en aanvalsstappen horen niet bij deze mindmap.

Veilig leren

Gebruik kennis over tooling om risico's te herkennen en verdediging te verbeteren; zet geen phishingproxy op.

5. Detectie van verdachte sessies en herkomst

- Let op ongebruikelijke locaties of netwerken, afwijkende apparaten of browsers en onmogelijke reispatronen.
- Controleer onverwachte MFA-aanvragen en sessieactiviteit die niet bij normaal gebruik past.
- Beoordeel signalen in samenhang: één afwijking bewijst geen compromittering.

6. FIDO2/WebAuthn

- FIDO2/WebAuthn koppelt authenticatie cryptografisch aan de legitieme website-oorsprong.
- Een nagemaakte domeinnaam kan die authenticatie niet eenvoudig naar een andere oorsprong omleiden.
- Daarom is deze methode ontworpen om phishing-resistent te zijn.

7. Conditionele toegang

- Toegang kan afhangen van context, zoals identiteit, apparaatstatus, locatie of risicosignalen.
- Organisaties kunnen aanvullende controles vragen of toegang blokkeren bij verdachte omstandigheden.
- Gebruik dit als extra verdedigingslaag naast phishing-resistente authenticatie en monitoring.

8. Korte samenvatting

- AiTM kan een geldige aangemelde sessie onderscheppen.
- Herken afwijkende aanmeldingen en sessies; beoordeel signalen gezamenlijk.
- Beperk risico met FIDO2/WebAuthn, monitoring en contextafhankelijk toegangsbeleid.

DORA: digitale operationele weerbaarheid in de financiële sector

1. Wat is DORA en voor wie geldt het?

- EU-verordening 2022/2554 over digitale operationele weerbaarheid.
- Geldt voor aangewezen financiële entiteiten; de precieze toepassing hangt af van de entiteit en de regels.

2. ICT-risicobeheerraamwerk

- Breng ICT-risico's en belangrijke systemen in kaart.
- Bescherm systemen en gegevens; detecteer gebeurtenissen.
- Reageer op verstoringen, herstel diensten en leer van incidenten en tests.

3. ICT-gerelateerde incidentrapportage

- Leg processen vast voor detectie, classificatie, respons en herstel.
- Documenteer en rapporteer ernstige ICT-incidenten volgens de toepasselijke DORA-regels.
- Rapportagevereisten en termijnen hangen af van de toepasselijke regels.

4. Testen van digitale operationele weerbaarheid (TLPT)

- Test of systemen verstoringen kunnen weerstaan en ervan kunnen herstellen.
- TLPT gebruikt dreigingsinformatie en realistische scenario's om weerbaarheid te toetsen.
- TLPT geldt onder voorwaarden voor entiteiten binnen de toepasselijke reikwijdte.

5. Beheer van ICT-derdenrisico's

- Beoordeel aanbieders, afhankelijkheden en risico's.
- Leg contractuele afspraken vast en volg risico's tijdens de samenwerking.
- Plan continuïteit en eventueel vertrek; de financiële entiteit blijft verantwoordelijk voor haar ICT-risico's.

6. Informatiedeling tussen instellingen

- Financiële entiteiten mogen informatie over cyberdreigingen en dreigingsinformatie delen.
- Houd daarbij rekening met vertrouwelijkheid, gegevensbescherming en mededingingsregels.

7. Handhaving en sancties

- Toezicht en handhaving kunnen maatregelen en sancties omvatten.
- De toepassing hangt af van DORA en nationale regelingen; er is niet één universeel sanctiebedrag.

8. Korte samenvatting

- DORA versterkt de digitale weerbaarheid van de financiële sector.
- De kern: risico's beheren, incidenten afhandelen, testen, derdenrisico's beheersen en informatie delen.
- Toezicht en handhaving ondersteunen de naleving.

Binaire Exploitatietechnieken: Return-Oriented Programming (ROP)

1. Wat is ROP?

Return-Oriented Programming (ROP) is een techniek voor codehergebruik. In plaats van nieuwe code uit te voeren, laat een aanvaller de programmastroom door korte stukjes bestaande uitvoerbare code lopen. Zo'n stukje heet een gadget; het eindigt vaak met een retourinstructie.

- Gadgets kunnen waarden laden, gegevens verplaatsen of bestaande functies aanroepen.
- ROP is een concept om te begrijpen bij geheugenbeveiliging, geen veilige programmeertechniek.

2. Geheugen en stack

De stack bewaart onder meer gegevens die nodig zijn bij functieaanroepen. Een retouradres geeft aan waar de uitvoering normaal verdergaat.
Als een kwetsbaarheid de controle over zo'n adres mogelijk maakt, kan de programmastroom worden omgeleid.

- Een retouradres is onderdeel van de normale programmastroom.
- ROP steunt op omleiding van de programmastroom; de stack kan adressen en bijbehorende gegevens bevatten.

3. Hoe een keten werkt

Een ROP-keten ordent adressen van gadgets en bijbehorende gegevens, zodat de uitvoering van het ene bestaande codefragment naar het volgende gaat.
Dit schema is uitsluitend conceptueel.

- Omleiding van de programmastroom → gadget A → gadget B → bestaande functie.
- Stackgegevens leveren de volgende adressen en benodigde waarden.
- De precieze werking hangt af van de bestaande code en de programmastroom.

Abstract schema
Geen doelgerichte adressen, uitvoerbare stappen of instructies voor het compromitteren van systemen.

4. DEP/NX

Data Execution Prevention (DEP) beperkt uitvoering van code uit geheugenpagina's die als niet-uitvoerbaar zijn gemarkeerd. NX is een benaming voor de niet-uitvoerbare geheugenmachtiging in bepaalde omgevingen.

- DEP/NX helpt voorkomen dat gegevenspagina's als code worden uitgevoerd.
- De precieze implementatie verschilt per systeem.

5. Waarom ROP DEP/NX kan omzeilen

DEP/NX voorkomt op zichzelf niet dat code die al in uitvoerbare geheugenregio's staat, wordt hergebruikt. ROP gebruikt zulke bestaande codefragmenten in plaats van code uit een niet-uitvoerbare gegevenspagina uit te voeren.

- DEP/NX en ROP gaan over verschillende aspecten: uitvoering van geheugenpagina's tegenover hergebruik van bestaande code.
- DEP/NX alleen is daarom geen volledige verdediging tegen codehergebruik.

6. Beperkingen

Een ROP-keten is afhankelijk van bruikbare gadgets in uitvoerbare code én van voldoende controle over de programmastroom. Bescherming kan die voorwaarden bemoeilijken; geen enkele genoemde maatregel garandeert op zichzelf dat ROP onmogelijk is.

- Beschikbaarheid van passende codefragmenten is van belang.
- Controle over de programmastroom is een noodzakelijke factor.
- De uitkomst hangt af van de situatie en aanwezige beveiligingen.

7. Verdediging

Gebruik meerdere verdedigingslagen om kwetsbaarheden te voorkomen en misbruik moeilijker te maken.

- **ASLR**: varieert waar onderdelen van programma's in het geheugen staan.
- Stack-canaries: helpen bepaalde overschrijvingen van de stack te signaleren.
- Maatregelen voor integriteit van de programmastroom beperken ongewenste overdrachten.
- Veilige programmeerpraktijken en tijdige updates helpen kwetsbaarheden voorkomen of verhelpen.

In samenhang
Deze maatregelen vullen elkaar aan; geen enkele maatregel is op zichzelf een garantie tegen ROP.

8. Korte samenvatting

ROP hergebruikt korte fragmenten van bestaande uitvoerbare code om de programmastroom te sturen. DEP/NX beperkt uitvoering vanaf als niet-uitvoerbaar gemarkeerde geheugenpagina's, maar voorkomt op zichzelf geen hergebruik van code uit uitvoerbare regio's.

- Gadgets vormen de bouwstenen; de stack kan adressen en gegevens voor de keten bevatten.
- Begrijp de techniek op conceptueel niveau en combineer verdedigingsmaatregelen.

Cloudbeveiliging: Entra ID (Azure AD) — hybride identiteit & risico's

1. Wat is het

Microsoft Entra ID is Microsofts cloudservice voor identiteits- en toegangsbeheer. Azure AD is de voormalige naam. In een hybride omgeving werken identiteiten uit de lokale Active Directory samen met identiteiten in de cloud.

- Hybride identiteit verbindt lokaal identiteitsbeheer met cloudtoegang.
- Gebruikersidentiteit bepaalt wie iemand is; toegangsbeleid bepaalt waartoe die persoon toegang krijgt.

2. Hoe werkt het

Bij aanmelden wordt gecontroleerd welke identiteit wordt gebruikt en of die toegang mag krijgen. De manier waarop de identiteit wordt beheerd, bepaalt waar de verificatie plaatsvindt.

- **Beheerde identiteit:** Microsoft Entra ID verzorgt de verificatie.
- **Gefedereerde identiteit:** een aparte identiteitsprovider verzorgt de verificatie.
- **Synchronisatie** kopieert geselecteerde identiteitsgegevens; het is geen verificatiemethode.

3. Synchronisatie

Synchronisatie houdt geselecteerde identiteitsgegevens uit de lokale Active Directory beschikbaar in Microsoft Entra ID. Dit gebeurt vaak met Microsoft Entra Connect Sync of Cloud Sync.

- Voorbeeld: een lokaal aangemaakte gebruikersidentiteit verschijnt na synchronisatie ook in de cloud.
- Welke gegevens worden gesynchroniseerd, hangt af van de inrichting.
- Een synchronisatieserver en de bijbehorende beheerdersrechten zijn kritieke onderdelen.

4. Federatie

Federatie delegeert de verificatie aan een afzonderlijke identiteitsprovider. Microsoft Entra ID kan een aanmelding doorsturen naar de identiteitsprovider van de organisatie, bijvoorbeeld een lokale voorziening.

- De identiteitsprovider controleert de aanmeldgegevens.
- Microsoft Entra ID kan daarna de cloudtoegang afhandelen volgens het ingestelde beleid.
- Een onjuiste configuratie kan aanmeldingen verstoren of beveiligingsrisico's veroorzaken.

5. Aanvalsvectoren

Aanvallers proberen vaak misbruik te maken van zwakke accounts, overmatige rechten of onvoldoende beveiligde koppelingen. Herken deze risico's om ze te helpen voorkomen.

- Gestolen of hergebruikte wachtwoorden.
- Beheerdersaccounts met meer rechten dan nodig.
- Onvoldoende multifactor-authenticatie (MFA).
- Kwetsbare of verkeerd ingerichte synchronisatie- en federatievoorzieningen.
- Onvoldoende controle van aanmeldingen en wijzigingen.

6. Risico's

Een incident in de lokale omgeving kan gevolgen hebben voor gekoppelde cloudidentiteiten. De impact hangt onder meer af van de toegangsrechten, inrichting en beschikbare detectie- en herstelmaatregelen.

- Misbruik van een account kan leiden tot ongeoorloofde toegang.
- Een gecompromitteerde lokale identiteit of synchronisatie-server kan cloudidentiteiten in gevaar brengen.
- Zonder voldoende logging kunnen verdachte aanmeldingen of wijzigingen onopgemerkt blijven.
- Problemen met federatie of synchronisatie kunnen toegang onderbreken.

7. Bescherming

Beperk de mogelijkheden van accounts en bescherm de systemen die identiteiten verbinden. Controleer daarnaast regelmatig of aanmeldingen en wijzigingen passen bij normaal gebruik.

- Geef beheerders alleen de rechten die zij nodig hebben.
- Bescherm bevoorrechte accounts met MFA.
- Beveilig en beperk toegang tot de synchronisatie-infrastructuur.
- Controleer aanmeldings- en auditlogboeken op afwijkingen.
- Houd herstelprocedures beschikbaar en actueel.

8. Korte samenvatting

Hybride identiteit verbindt lokale Active Directory met Microsoft Entra ID. Synchronisatie kopieert geselecteerde identiteitsgegevens; federatie delegeert verificatie. Beide koppelingen vragen om zorgvuldig beheer en bewaking.

- Beperk rechten en bescherm beheerdersaccounts met MFA.
- Beveilig de synchronisatie- en federatievoorzieningen.
- Controleer logboeken en wees voorbereid op herstel.

Cross-Site WebSocket Hijacking (CSWSH)

Een risico waarbij een kwaadwillende website misbruik probeert te maken van de aangemelde browsersessie van een gebruiker om een WebSocket-verbinding met een andere website te openen.

1 Wat is het

CSWSH is een risico op cross-site request forgery (CSRF) rond de WebSocket-handshake.

- De browser kan toepasselijke cookies automatisch meesturen.
- Zonder controle van de **herkomst** kan een ongewenste site een verbinding proberen te openen in de context van de gebruiker.

2 Hoe werkt WebSocket

WebSocket maakt na een openingsfase een blijvende, tweerichtingsverbinding mogelijk.

- Eerst vraagt de client via een HTTP-handshake om de verbinding te openen.
- Na acceptatie kunnen client en server berichten naar elkaar sturen.
- De handshake en de daaropvolgende gegevensuitwisseling zijn verschillende fasen.

3 Hoe ontstaat CSWSH

Het risico ontstaat als een server een WebSocket-handshake accepteert op basis van alleen een ingelogde sessie en de herkomst niet controleert.

- De browser kan cookies meesturen die bij de doelsite horen.
- De server controleert de Origin-waarde niet tegen vertrouwde herkomsten.
- De verbinding kan daardoor namens de ingelogde gebruiker tot stand komen.

4 Voorbeeldscenario

Een gebruiker is aangemeld bij een chatdienst. Een kwaadwillende webpagina probeert vanuit de browser een WebSocket-verbinding met die dienst te openen.

- Als de server de sessie vertrouwt maar de Origin niet controleert, kan de handshake worden geaccepteerd.
- Een veilige server weigert de handshake wanneer de Origin niet op de vertrouwde lijst staat.

5 Gevolgen en risico's

Wat er mis kan gaan, hangt af van de bevoegdheden van de verbinding en de controles op berichten en handelingen.

- Een onbevoegde verbinding kan toegang geven tot gegevens of acties binnen de toepassing.
- Onvoldoende autorisatie per handeling vergroot het risico.
- Een aangemelde sessie alleen bewijst niet dat de verbinding van een vertrouwde website komt.

6 Herkennen en testen

Beoordeel het ontwerp en de configuratie op hoofdlijnen; voer geen aanval uit op systemen zonder toestemming.

- Controleer conceptueel of de server de Origin tijdens de handshake valideert.
- Ga na of gevoelige handelingen afzonderlijk worden geautoriseerd.
- Browserbeveiliging rond dezelfde herkomst vervangt servercontroles niet.

7 Bescherming

Combineer herkomstcontrole met passende beveiligingsmaatregelen.

- Valideer de Origin tijdens de handshake tegen een expliciete lijst van toegestane herkomsten.
- Gebruik passende authenticatie en autoriseer elke gevoelige handeling.
- Vertrouw niet uitsluitend op cookies als bewijs van een vertrouwde herkomst.
- Gebruik wss:// om verkeer tijdens transport te beschermen.
- Origin-controle vervangt authenticatie, autorisatie of bescherming tegen niet-browserclients niet.

Onthoud
Controleer de herkomst én de bevoegdheid. Geen enkele controle vervangt alle andere.

8 Korte samenvatting

CSWSH misbruikt mogelijk een aangemelde browsersessie tijdens het openen van een WebSocket-verbinding.

- De handshake is de openingsfase; daarna volgt tweerichtingsverkeer.
- Valideer de Origin en combineer dit met authenticatie en autorisatie.
- Gebruik wss:// voor versleuteling van het verkeer tijdens transport.

Kubernetes-beveiliging: RBAC, Pod Security Standards & toelating

1. Wat is het

- Beveilig zowel wie Kubernetes-acties mag uitvoeren als welke workloads aan beleid voldoen.
- RBAC regelt API-toegang; admission beoordeelt aanvragen vóór objectopslag.

2. RBAC

- Een Role beschrijft toegestane API-groepen, resources en werkwoorden, zoals get, list, create en update.
- Een RoleBinding koppelt rechten binnen één namespace aan een gebruiker, groep of serviceaccount.
- ClusterRole en ClusterRoleBinding kunnen rechten op clusterniveau toekennen. Een ClusterRole kan ook via bindings in namespaces worden gebruikt.
- RBAC verleent rechten; het is geen expliciet deny-mechanisme.
- Pas minimale toegang toe, vermijd onnodige wildcards en controleer serviceaccountrechten.

3. Pod Security Standards

- De profielen voor podconfiguraties zijn Privileged, Baseline en Restricted: van ruim permissief naar sterker beperkt.
- Pod Security Admission (PSA) past gekozen profielen toe op namespace-niveau.
- De modi zijn enforce (weigeren), audit (registreren) en warn (waarschuwen).
- De labels pod-security.kubernetes.io/enforce, /audit en /warn kunnen elk een profiel en optionele versie aangeven.

4. Admission Controllers

- Admission controllers beoordelen API-aanvragen na authenticatie en autorisatie, maar vóór opslag van het object.
- Ze kunnen aanvragen valideren of muteren; webhooks kunnen eigen validatie of mutatie toevoegen.
- Voorbeeld: een validatieregel weigert een workload die privileged containers aanvraagt.
- Voorbeeld: een mutatieregel voegt standaardlabels toe. Dit zijn illustraties, geen gegarandeerde standaard-configuraties.
- PSA is een specifieke toepassing van admission voor Pod Security Standards; niet elke admission controller handhaaft die standaarden.

5. Hoe werken ze samen

- RBAC beperkt wie bepaalde API-acties mag uitvoeren.
- Admission beoordeelt of een aanvraag aan beleidsregels voldoet; PSA past daarbij gekozen podbeveiligingsprofielen toe.
- Samen bieden ze toegangscontrole en beleidscontrole op aanvragen.
- Ze vervangen geen netwerkbeleid, imagebeveiliging of runtimebewaking.

6. Praktijkvoorbeeld

- Een team mag workloads aanmaken in de eigen namespace via een passende Role en RoleBinding.
- PSA kan daar een gekozen Pod Security-profiel toepassen.
- Een aanvraag met een niet-toegestane privileged configuratie kan worden geweigerd wanneer het beleid daarop is ingesteld.
- Test beleidswijzigingen waar passend eerst met audit- of waarschuwingsmodus.

7. Veelgemaakte fouten

- Meer rechten geven dan nodig, bijvoorbeeld met ongerichte wildcards.
- Vergeten te controleren welke rechten serviceaccounts hebben.
- Aannemen dat RBAC zelf onveilige podconfiguraties afkeurt.
- Een profiel kiezen zonder de gevolgen voor workloads te beoordelen.
- Denken dat admission andere beveiligingsmaatregelen overbodig maakt.

8. Korte samenvatting

- RBAC: wie mag welke API-acties uitvoeren?
- Pod Security Standards: welke podconfiguraties passen bij het gekozen profiel?
- Admission: aanvragen valideren of muteren vóór opslag; PSA past Pod Security Standards toe.
- Beperk rechten, kies een passend profiel en test beleidswijzigingen zorgvuldig.

Versleutelde DNS: DoH en DoT

1. Wat is versleutelde DNS?

DNS vertaalt domeinnamen naar DNS-records, bijvoorbeeld een IP-adres. Traditionele DNS-verzoeken zijn doorgaans niet versleuteld tussen jouw apparaat en de resolver: de server die het verzoek opzoekt.

- DoH en DoT versleutelen DNS-verkeer onderweg naar de resolver.
- De resolver verwerkt het verzoek; kies daarom een resolver die je vertrouwt.

2. Hoe werkt DoH?

DNS over HTTPS (DoH) verstuurt DNS-berichten via HTTPS, doorgaans over poort 443.

- Het DNS-verzoek gaat mee in een HTTPS-verbinding naar een resolver.
- Voorbeelden van instellingen zijn een beheerde resolver in een browser of op een apparaat.

3. Hoe werkt DoT?

DNS over TLS (DoT) versleutelt DNS-verkeer met TLS, doorgaans via poort 853.

- Het apparaat maakt een versleutelde verbinding met een resolver die DoT ondersteunt.
- De verbinding beschermt het verzoek onderweg; de resolver ontvangt en verwerkt het.

4. Privacyvoordelen

Versleuteling maakt het voor tussenliggende netwerkwaarnemers lastiger om de inhoud van DNS-verzoeken mee te lezen.

- De DNS-inhoud is tijdens transport afgeschermd.
- De gekozen resolver kan het verzoek nog steeds zien.
- Andere metadata, zoals verbindingen met IP-adressen, kunnen zichtbaar blijven.

5. Beveiligingsvoordelen

Een correct ingestelde versleutelde verbinding helpt DNS-verzoeken onderweg te beschermen tegen eenvoudig meelezen en manipulatie op het netwerkpad.

- DoH gebruikt HTTPS; DoT gebruikt TLS.
- De bescherming hangt af van de juiste configuratie en het vertrouwen in de resolver.
- Versleutelde DNS voorkomt niet elk beveiligingsrisico.

6. Impact op bedrijfszichtbaarheid

DoH en DoT kunnen traditionele DNS-monitoring in een bedrijfsnetwerk lastiger maken. Hoeveel zichtbaarheid overblijft, hangt af van de configuratie en beschikbare controlepunten.

- Centrale observatie, filtering of het correleren van domeinverzoeken kan moeilijker worden.
- Versleuteling van DNS betekent niet dat alle netwerkactiviteit onzichtbaar is.
- Houd rekening met zowel privacy als de behoefte aan beveiligingsmonitoring.

7. Beheer en afwegingen

Organisaties kunnen zelf kiezen hoe zij DNS-verkeer beheren, bijvoorbeeld met beheerde resolvers, apparaatinstellingen en netwerkbeleid.

- Bepaal welke resolvers zijn toegestaan en hoe apparaten daarop worden ingesteld.
- Breng in kaart waar DNS-verkeer zichtbaar of controleerbaar moet zijn.
- Stem privacy, beveiliging en operationele behoeften op elkaar af.

Praktische afweging

Een versleuteld DNS-kanaal beschermt het verkeer onderweg, maar neemt de noodzaak van een betrouwbare resolver en passende monitoring niet weg.

8. Korte samenvatting

DoH en DoT versleutelen DNS-verzoeken tijdens transport naar een resolver. Dat kan meelezen en manipulatie onderweg bemoeilijken, maar de resolver blijft een vertrouwenspunt en bedrijfsmonitoring kan minder zicht op DNS-verzoeken krijgen.

- DoH: DNS via HTTPS, doorgaans poort 443.
- DoT: DNS met TLS, doorgaans poort 853.
- Bescherming onderweg is geen garantie op volledige anonimiteit of beveiliging.

Mindmap 133: Geheugenforensica met Volatility Framework

Geheugenforensica met Volatility Framework

1. Wat is het

Geheugenforensica onderzoekt een vastgelegde kopie van het vluchtige RAM-geheugen. Volatility Framework helpt onderzoekers die gegevens te analyseren.

- Zoek naar aanwijzingen over processen, geheugen en netwerkactiviteit.
- Uitkomsten hangen af van het besturingssysteem, de geheugenimage en de beschikbare analysemogelijkheden.

2. Waarom vluchtig geheugen

RAM kan sporen bevatten die verdwijnen wanneer een systeem wordt uitgeschakeld. Een geheugenimage is een momentopname, geen volledig verslag van wat er ooit gebeurde.

- Kan actieve processen en bepaalde netwerkverbindingen tonen.
- Leg vast wanneer en onder welke omstandigheden de image is gemaakt.

3. Voorbereiding en integriteit

Begin met zorgvuldige veiligstelling en documentatie. Bewaar het origineel ongewijzigd en analyseer waar mogelijk een werkkopie.

- Noteer herkomst, tijdstip en relevante omstandigheden.
- Registreer integriteitshashes om bestanden later te kunnen vergelijken.
- Identificeer het vermoedelijke besturingssysteem en controleer welke symbolen en analysemogelijkheden passen.
- Leg gebruikte instellingen, bevindingen en beperkingen reproduceerbaar vast.

4. Processen en verborgen processen

Procesoverzichten en onderlinge relaties kunnen afwijkingen zichtbaar maken. Vergelijk meerdere informattebronnen; een afwijking is een aanwijzing om verder te onderzoeken, geen zelfstandig bewijs van malware.

- Let op onverwachte relaties of ontbrekende vermeldingen.
- Controleer of de bevinding past bij het systeem en de context.

5. Geheugenregio's en code-injectie

Analyse van geheugenregio's en uitvoerbare code kan aanwijzingen geven voor mogelijke code-injectie. Een ongebruikelijke geheugenindeling bewijst op zichzelf geen kwaadaardige activiteit.

- Vergelijk de waarneming met andere bevindingen en de systeemcontext.
- Hypothetisch voorbeeld: een onverwachte uitvoerbare geheugenregio is reden voor nader onderzoek, maar geen bewijs van injectie.

6. Netwerkverbindingen

Netwerkanalyse kan verbindingen en eindpunten tonen die in de geheugenimage zijn vastgelegd. De momentopname geeft niet de volledige netwerkgeschiedenis weer.

- Beoordeel verbindingen in relatie tot processen en het tijdstip van vastlegging.
- Beschouw ontbrekende gegevens niet automatisch als bewijs dat er geen activiteit was.

7. Interpretatie en verslaglegging

Behandel resultaten als contextuele aanwijzingen en vergelijk waar mogelijk verschillende bronnen. Noteer wat is onderzocht, wat is gevonden en wat onzeker blijft.

- Vermeld gebruikte analysecategorieën, parameters en relevante instellingen.
- Beschrijf beperkingen van de image en van de analyse.
- Plugin-namen en beschikbaarheid verschillen per Volatility-versie; vermijd versieafhankelijke aannames.

8. Korte samenvatting

Volatility ondersteent de analyse van vastgelegd RAM. Zorgvuldige bewaring, vergelijking van aanwijzingen en heldere verslaglegging helpen overhaaste conclusies voorkomen.

- Werk op een kopie en documenteer herkomst en integriteit.
- Interpreteer proces-, geheugen- en netwerkbevindingen in context.
- Een afwijking is een aanleiding tot nader onderzoek, niet automatisch bewijs.

Privacy en gegevensbescherming: pseudonimisering, anonimisering en PET's

1. Begrippen

Privacybeschermende technieken helpen risico's voor persoonsgegevens te beperken. Pseudonimisering en anonimisering zijn verschillende manieren om gegevens minder direct herkenbaar te maken.

- Persoonsgegevens zijn gegevens die over een geïdentificeerde of identificeerbare persoon gaan.
- Pseudonimisering vervangt directe identificerende gegevens door een code of andere aanduiding.
- Anonimisering beoogt identificatie met redelijkerwijs waarschijnlijke middelen onmogelijk te maken.

2. Pseudonimisering

Bij pseudonimisering blijft een koppeling met een persoon mogelijk via aanvullende informatie. Behandel de gegevens daarom zorgvuldig.

- Vervang bijvoorbeeld klantnamen door willekeurige codes.
- Bewaar de koppeltabel afzonderlijk en beperk de toegang ertoe.
- Gepseudonimiseerde gegevens kunnen nog steeds persoonsgegevens zijn.

3. Anonimisering

Anonimisering is bedoeld om personen niet langer identificeerbaar te maken met middelen die redelijkerwijs waarschijnlijk kunnen worden gebruikt.

- Alleen namen verwijderen is geen garantie voor anonimisering.
- Indirecte kenmerken, zoals een zeldzame functie of combinatie van kenmerken, kunnen iemand herkenbaar maken.
- Bij daadwerkelijk geanonimiseerde gegevens is koppeling niet redelijkerwijs herstelbaar.

4. Privacybevorderende technologieën (PETs)

PETs zijn technieken die kunnen helpen privacyrisico's te beperken. Kies maatregelen passend bij de gegevens en het gebruik.

- Versleuteling maakt gegevens zonder passende sleutel moeilijk leesbaar.
- Pseudonimisering vervangt directe identificerende gegevens door codes.
- Gegevensminimalisatie beperkt welke gegevens worden verzameld of gebruikt.
- Toegangsbeperking en privacyvriendelijke statistische analyse zijn andere voorbeelden.

5. Voorbeelden

Een analysetabel bevat klantcodes in plaats van klantnamen. De organisatie bewaart de koppeltabel apart en geeft alleen bevoegde medewerkers toegang. Dit is pseudonimisering, geen anonimisering.

- Versleuteling beschermt de leesbaarheid, maar maakt gegevens niet automatisch anoniem.
- Een tabel zonder namen kan nog herkenbaar zijn door kleine groepen of zeldzame kenmerken.
- Gegevens combineren met andere bronnen kan herkenning mogelijk maken.

6. Voordelen

Passende privacymaatregelen kunnen blootstelling van gegevens verminderen en de toegang beter beheersbaar maken.

- Minder directe identificatie in analyses of testomgevingen.
- Beperkte toegang tot de informatie die koppeling mogelijk maakt.
- Minder verzamelde gegevens kunnen het privacyrisico verlagen.

7. Grenzen en risico's

Geen enkele techniek neemt elk privacyrisico weg. Beoordeel gegevens, context en mogelijke combinaties als geheel.

- Kleine groepen en zeldzame kenmerken kunnen personen herkenbaar maken.
- Een code is geen anonimisatie wanneer een koppeltabel bestaat.
- Versleutelde gegevens kunnen persoonsgegevens blijven als ze aan iemand te koppelen zijn.
- Denk aan doelbinding, gegevensminimalisatie, passende beveiliging en heridentificatierisico; een techniek op zichzelf garandeert geen naleving.

Let op
Verwijderde namen, codes of versleuteling bewijzen op zichzelf niet dat gegevens anoniem zijn.

8. Korte samenvatting

Pseudonimisering vermindert directe herkenbaarheid, maar laat een koppelings-mogelijkheid bestaan. Anonimisering beoogt die mogelijkheid redelijkerwijs weg te nemen. PETs helpen privacyrisico's te beperken, maar vragen om een beoordeling van de context.

- Pseudonimisering: code plus apart beschermde koppelgegevens.
- Anonimisering: personen zijn niet redelijkerwijs identificeerbaar.
- Versleuteling en namen verwijderen zijn niet automatisch anonimisering.

Cyberweerbaarheid: crisiscommunicatie & stakeholdermanagement

1. Wat is het?

- Planmatig informatie delen tijdens en na een cyberincident.
- Geef juiste, tijdige en begrijpelijke informatie.

2. Waarom communicatie ertoe doet

- Beperk verwarring en help betrokkenen passende acties te nemen.
- Corrigeer onjuiste informatie en stem interne en externe berichten op elkaar af.

3. Stakeholders en behoeften

- Denk aan medewerkers, klanten, leveranciers, bestuur, toezichthouders en hulpverleners.
- Stem inhoud en kanaal af op elke doelgroep.

4. Rollen en voorbereiding

- Wijs rollen toe voor coördinatie, communicatie, juridische beoordeling en goedkeuring.
- Houd contactgegevens, escalatiepaden en berichtsjablonen actueel.

5. Communicatie tijdens het incident

- Meld wat bekend is, wat nog wordt onderzocht en welke acties nodig zijn.
- Geef aan wanneer de volgende update volgt.
- Leg besluiten, tijdstippen, feiten en verzonden berichten vast.
- Deel gevoelige informatie alleen via passende kanalen met bevoegde personen; vermijd speculatie en onnodige verspreiding van persoonsgegevens.

6. Juridische aandachtspunten

- Meld- en informatieplichten hangen af van jurisdictie, sector, incident en betrokken gegevens.
- Breng toepasselijke verplichtingen in kaart en laat ze juridisch beoordelen; ga niet uit van één universele termijn.

Let op
Deze mindmap is educatief en geen juridisch advies.

7. Herstel en evaluatie

- Evalueer de communicatie en verzamel feedback van betrokkenen.
- Werk plannen, contactgegevens en berichtsjablonen bij.

8. Korte samenvatting

- Communiceer tijdig, begrijpelijk en feitelijk.
- Stem berichten af op doelgroep, rol en kanaal.
- Leg vast, bescherm gevoelige informatie en leer na afloop.

Detectie en respons op identiteitsdreigingen (ITDR)

1. Wat is ITDR

- ITDR helpt dreigingen tegen accounts, aanmeldgegevens en identiteitsinfrastructuur te herkennen en beheersen.
- Identiteitssystemen regelen wie iemand is, hoe die zich aanmeldt en waartoe die toegang heeft.
- Doel: verdachte activiteit onderzoeken en risico zorgvuldig beperken.

2. Kwetsbaarheden in identiteitssystemen

- Zwakke of hergebruikte wachtwoorden vergroten het risico op accountmisbruik.
- Te ruime toegangsrechten geven een account meer mogelijkheden dan nodig.
- Onbeschermde accounts en onveilige configuraties verdienen extra aandacht.
- Voorbeeld: een oud account met toegangsrechten blijft actief nadat iemand van rol is veranderd.

3. Realtime gedragsanalyse van accounts

- Vergelijk activiteiten met gebruikelijke patronen: tijdstip, locatie, apparaat en toegangsbron.
- Een nieuwe locatie of ongebruikelijk tijdstip is een signaal om te onderzoeken, geen bewijs van misbruik.
- Bekijk context, zoals gepland werk of een nieuw apparaat, voordat je conclusies trekt.

Let op
Een afwijking kan een legitieme verklaring hebben. Beoordeel meerdere signalen en de context.

4. Detectie van misbruik van aanmeldgegevens

- Aanmeldgegevens zijn bijvoorbeeld wachtwoorden en tokens.
- Mogelijke signalen: ongebruikelijke aanmeldingen, herhaalde mislukte pogingen of onverwachte toegang tot bronnen.
- Voorbeeld: een account meldt zich op een ongewoon moment aan en opent daarna bestanden die het normaal niet gebruikt.
- Controleer of de activiteit past bij de gebruiker en diens werk.

5. Bescherming van Active Directory en IdP's

- Active Directory en identiteitsproviders (IdP's) ondersteunen centraal identiteitsbeheer.
- Beperk rechten tot wat nodig is en bescherm beheeraccounts extra zorgvuldig.
- Gebruik sterke authenticatie en bewaak wijzigingen aan accounts en toegangsrechten.
- Voorbeeld: onderzoek onverwachte wijzigingen aan een beheerdersaccount.

6. Geautomatiseerde responsacties (beperking)

- Mogelijke maatregelen: een account tijdelijk blokkeren of beperken, sessies of tokens intrekken en extra verificatie vragen.
- Stem de actie af op de ernst van het signaal en de mogelijke gevolgen voor het werk.
- Leg vast wat is aangepast en controleer of de dreiging is beperkt.

Reageer zorgvuldig
Beperk bevestigde of sterk vermoede dreigingen, maar houd rekening met de gevolgen voor de gebruiker en bedrijfsprocessen.

7. Integratie met SIEM, SOAR en EDR

- SIEM verzamelt en correleert beveiligingsgebeurtenissen uit verschillende bronnen.
- SOAR ondersteunt het coördineren of automatiseren van responsprocessen.
- EDR richt zich op detectie en respons op eindpunten, zoals computers en servers.
- Integratie kan signalen en context delen, zodat een aanmeldsignaal samen met eindpuntactiviteit kan worden onderzocht.

8. Korte samenvatting

- Bewaak identiteitsactiviteiten en let op afwijkingen.
- Onderzoek signalen in context; afwijkend gedrag is niet automatisch misbruik.
- Beperk bevestigde of sterk vermoede dreigingen zorgvuldig.
- Gebruik bevindingen om beveiliging en respons te verbeteren.

Red Teaming en Purple Teaming: methodologieën

1 Definities: Red, Blue en Purple

De teams hebben verschillende rollen, maar werken aan hetzelfde doel: de weerbaarheid verbeteren.

- **Red Team:** simuleert afgesproken adversarygedrag om preventie, detectie en respons te toetsen.
- **Blue Team:** verdedigt, monitort, onderzoekt waarschuwingen en reageert op incidenten.
- **Purple Team:** laat Red- en Blue Team inzichten delen en samen verbeteringen testen.

2 Doelstellingen van adversary-emulatie

Emulatie bootst tactieken, technieken en procedures van een gekozen dreigingsprofiel na om beveiligingsmaatregelen en detecties te toetsen.

- Kies vooraf wat je wilt leren, zoals of een bepaalde activiteit wordt opgemerkt.
- Een gekozen profiel is een testlens, geen dekking van alle mogelijke aanvallen.
- Leg vast welke systemen en processen binnen de test vallen.

3 Scenariokaders: TIBER-EU en MITRE ATT&CK

Gebruik kaders passend bij hun doel, ze zijn niet onderling uitwisselbaar.

- **MITRE ATT&CK** is een kennisbank voor adversary-tactieken en -technieken. Een mapping helpt tests te structureren, maar is geen formele scenariostandaard of volledigheidsgarantie.
- **TIBER-EU** is een toepassingsgericht raamwerk voor gecontroleerde, dreigingsgebaseerde weerbaarheidstests bij financiële instellingen en kritieke functies; het is geen universele norm.

4 Uitveoring van gesimuleerde aanvallen

Voer uitsluitend een vooraf goedgekeurde, begrensde simulatie uit.

- Leg doelen, scope, toestemming, regels, stopcriteria en contactpersonen vast.
- Spreek af hoe productie-impact en blootstelling van gegevens worden beperkt.
- Registreer teststappen en waarnemingen; stop of escaleer bij onverwachte impact volgens de afgesproken criteria.

Voorbeeld

Een testplan benoemt welke omgeving binnen scope valt en wie wordt gebeld als de test onverwachte verstoring veroorzaakt.

5 Realtime samenwerking en kennisoverdracht

Purple Teaming maakt bevindingen bruikbaar voor zowel testen als verdediging.

- Bespreek waarnemingen en relevante indicatoren binnen de afgesproken vertrouwelijkheid.
- Loop samen na welke logging, waarschuwingen en analistenreacties zichtbaar waren.
- Blijf binnen de testgrenzen; deel alleen informatie die voor de evaluatie nodig is.

6 Detectievalidatie en tuning

Vergelijk wat de test deed met wat systemem en analisten daadwerkelijk waarnamen.

- Controleer logging, waarschuwingen en opvolging door analisten.
- Noteer detectiegaten en mogelijke oorzaken, zoals ontbrekende telemetrie of een te ruime regel.
- Pas detectieregels gericht aan en controleer of ongewenste meldingen niet onnodig toenemen.

7 Rapportage en voortdurende verbetering

Maak de uitkomst navolgbaar en zet bevindingen om in geprioriteerde acties.

- Rapporteer doel en scope, geteste scenario's, waarnemingen, detectiegaten en beperkingen.
- Wijs verbeteracties toe en herhaal tests om te beoordelen of wijzigingen effect hebben.
- Voorbeeld: na een gemiste waarschuwing verbeteren teams samen een detectieregel en toetsen die opnieuw.

8 Korte samenvatting

Gecontroleerde simulatie levert de meeste leerwaarde op wanneer verdedigers en testers bevindingen samen omzetten in toetsbare verbeteringen.

- Red simuleert; Blue verdedigt; Purple verbindt inzichten.
- Emuleer een gekozen profiel binnen expliciete grenzen; claim geen volledige dekking.
- Gebruik ATT&CK om technieken te ordenen en TIBER-EU als specifiek toepassingskader.
- Valideer detecties, rapporteer beperkingen en herhaal de test.

Mindmap 147: Cyberverzekeringen en kwantificering van cyberrisico's (FAIR-model)

1. Het FAIR-framework

FAIR staat voor Factor Analysis of information Risk. Dit raamwerk helpt informatiebeveiligingsrisico's in financiële termen te analyseren.

- Ordent risico met twee kernbegrippen: frequentie van verliesgebeurtenissen (LEF) en omvang van verlies (LM).
- De uitkomst is een schatting, geen zekerheid.

2. Frequentie van Verliesgebeurtenissen (Loss Event Frequency)

LEF schat hoe vaak een verliesgebeurtenis binnen een gekozen periode kan optreden.

- Hangt samen met hoe vaak dreigingsgebeurtenissen voorkomen en hoe waarschijnlijk die tot verlies leiden.
- Voorbeeld: schat hoe vaak een accountincident in een jaar tot verstoring of kosten kan leiden.
- Noteer de periode, aannames en onzekerheid; een schatting voorspelt geen exacte uitkomst.

3. Omvang van Verlies (Loss Magnitude)

LM schat de financiële omvang van het verlies per gebeurtenis.

- Denk aan directe herstel- en responskosten en mogelijke vervolgkosten, zoals bedrijfsstilstand of aansprakelijkheid.
- Voorbeeld: bij een hypothetisch incident kunnen herstelwerk en tijdelijk omzetverlies allebei meetellen.
- Maak aannames zichtbaar en vermijd schijnprecisie: werkelijke kosten hangen af van de situatie.

4. Primaire vs Secundaire Schade

Deze indeling helpt verschillende gevolgen van een incident te rubriceren.

- Primaire schade treft de organisatie rechtstreeks, bijvoorbeeld door herstelkosten of onderbreking van bedrijfsactiviteiten.
- Secundaire schade kan ontstaan door reacties van externe partijen, zoals klanten, toezichthouders of zakenpartners.
- Mogelijke gevolgen zijn juridische kosten, boetes waar van toepassing, reputatieschade of omzetverlies. Categorie en omvang verschillen per situatie.

5. Eisen voor Cyberverzekeringen

Bij een aanvraag of beoordeling kan een verzekeraar informatie vragen over de organisatie, beveiligingsmaatregelen en het te verzekeren risico.

- Aanvraagvereisten en acceptatiecriteria verschillen per verzekeraar en polis.
- Gebruik eventuele vragen om risico's en maatregelen in kaart te brengen; ze vormen geen universele bindende checklist.
- Controleer de voorwaarden van de specifieke polis.

6. Uitsluitingen & Clausules

Polisvoorwaarden beschrijven onder meer dekking, limieten, eigen risico, meldingsverplichtingen en uitsluitingen.

- Mogelijke aandachtspunten zijn oorlogshandelingen, bekende incidenten of bepaalde systeem- en beveiligingsomstandigheden.
- Formulering en toepasselijkheid verschillen per polis en jurisdictie.
- Een voorbeeld garandeert niet dat iets wel of niet gedekt is; lees de concrete voorwaarden.

Let op

Deze mindmap is educatief en geen juridisch advies. Trek geen conclusie over dekking zonder de toepasselijke polisvoorwaarden te lezen.

7. Risico-overdracht versus Mitigatie

Verzekering en beveiligingsmaatregelen kunnen verschillende rollen spelen bij risicobeheer.

- Verzekering kan financiële gevolgen van bepaalde gedekte verliezen helpen verdelen; niet elk verlies is overdraagbaar of verzekerd.
- Mitigatie verlaagt de kans op of impact van verlies, bijvoorbeeld met toegangsbeheer, back-ups en incidentvoorbereiding.
- Verzekering vervangt beveiligingsmaatregelen niet.

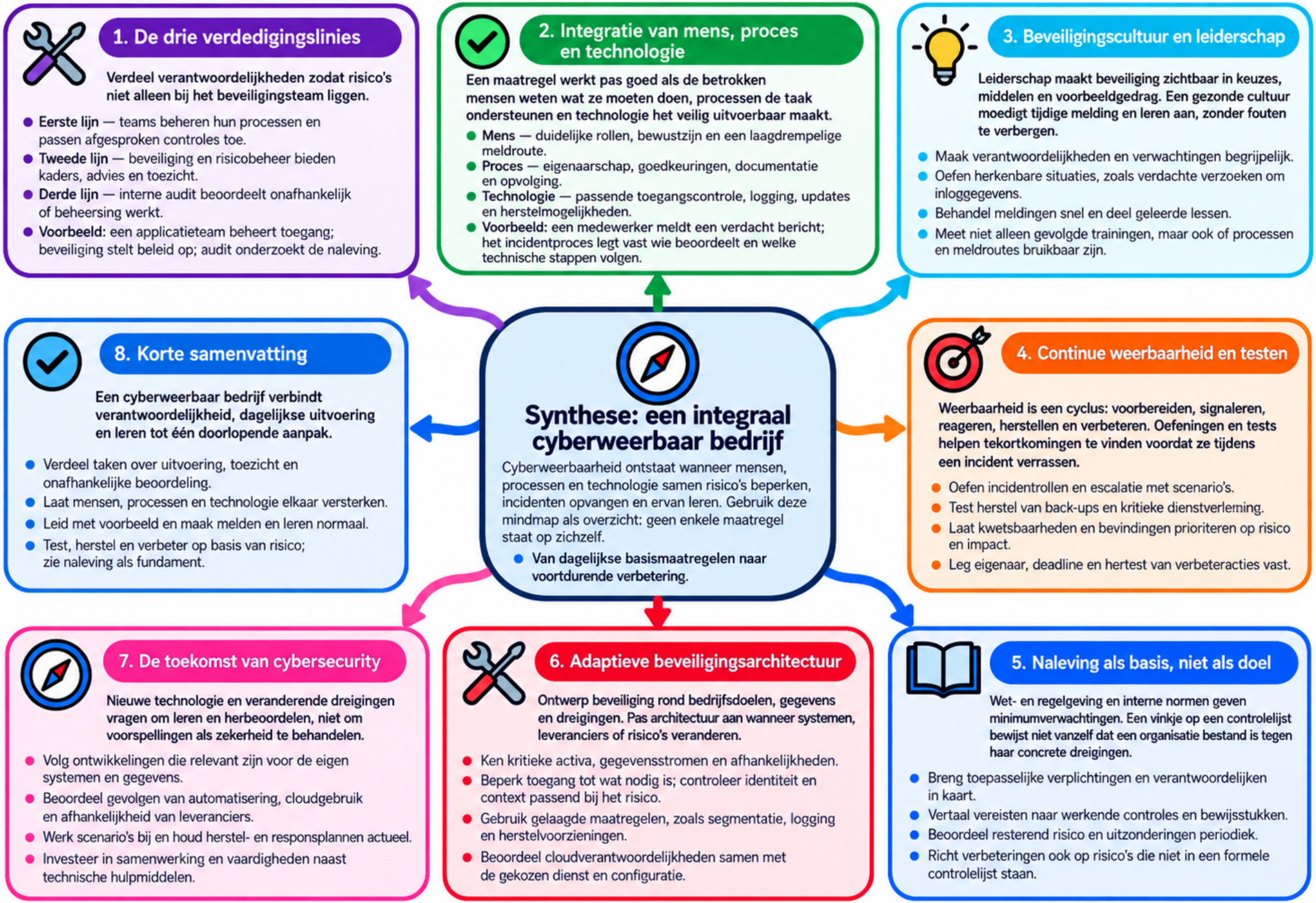
8. Korte samenvatting

FAIR ordent cyberrisico als geschatte frequentie en financiële impact. Verliescategorieën verduidelijken mogelijke gevolgen; polisvoorwaarden begrenzen mogelijke dekking en mitigatie blijft nodig.

- LEF: hoe vaak kan een verliesgebeurtenis optreden?
- LM: hoe groot kan het financiële verlies per gebeurtenis zijn?
- Maak aannames expliciet: schattingen zijn onzeker en polisvoorwaarden verschillen.

Onthoud

Een hypothetische risico-inschatting is geen voorspelling en zegt op zichzelf niets definitiefs over verzekeringsdekking.


1. De drie verdedigingslinies
Verdeel verantwoordelijkheden zodat risico's niet alleen bij het beveiligingsteam liggen.
Eerste lijn — teams beheren hun processen en passen afgesproken controles toe.
Tweede lijn — beveiliging en risicobeheer bieden kaders, advies en toezicht.
Derde lijn — interne audit beoordeelt onafhankelijk of beheersing werkt.
Voorbeeld: een applicatieteam beheert toegang; beveiliging stelt beleid op; audit onderzoekt de naleving.
2. Integratie van mens, proces en technologie
Een maatregel werkt pas goed als de betrokken mensen weten wat ze moeten doen, processen de taak ondersteunen en technologie het veilig uitvoerbaar maakt.
Mens — duidelijke rollen, bewustzijn en een laagdrempelige meldroute.
Proces — eigenaarschap, goedkeuringen, documentatie en opvolging.
Technologie — passende toegangscontrole, logging, updates en herstelmogelijkheden.
Voorbeeld: een medewerker meldt een verdacht bericht; het incidentproces legt vast wie beoordeelt en welke technische stappen volgen.
3. Beveiligingscultuur en leiderschap
Leiderschap maakt beveiliging zichtbaar in keuzes, middelen en voorbeeldgedrag. Een gezonde cultuur moedigt tijdige melding en leren aan, zonder fouten te verbergen.
Maak verantwoordelijkheden en verwachtingen begrijpelijk.
Oefen herkenbare situaties, zoals verdachte verzoeken om inloggegevens.
Behandel meldingen snel en deel geleerde lessen.
Meet niet alleen gevolgde trainingen, maar ook of processen en meldroutes bruikbaar zijn.
8. Korte samenvatting
Een cyberweerbaar bedrijf verbindt verantwoordelijkheid, dagelijkse uitvoering en leren tot één doorlopende aanpak.
Verdeel taken over uitvoering, toezicht en onafhankelijke beoordeling.
Laat mensen, processen en technologie elkaar versterken.
Leid met voorbeeld en maak melden en leren normaal.
Test, herstel en verbeter op basis van risico; zie naleving als fundament.
Synthese: een integraal cyberweerbaar bedrijf
Cyberweerbaarheid ontstaat wanneer mensen, processen en technologie samen risico's beperken, incidenten opvangen en ervan leren. Gebruik deze mindmap als overzicht: geen enkele maatregel staat op zichzelf.
Van dagelijkse basismaatregelen naar voortdurende verbetering.
4. Continue weerbaarheid en testen
Weerbaarheid is een cyclus: voorbereiden, signaleren, reageren, herstellen en verbeteren. Oefeningen en tests helpen tekortkomingen te vinden voordat ze tijdens een incident verrassen.
Oefen incidentrollen en escalatie met scenario's.
Test herstel van back-ups en kritieke dienstverlening.
Laat kwetsbaarheden en bevindingen prioriteren op risico en impact.
Leg eigenaar, deadline en hertest van verbeteracties vast.
7. De toekomst van cybersecurity
Nieuwe technologie en veranderende dreigingen vragen om leren en herbeoordelen, niet om voorspellingen als zekerheid te behandelen.
Volg ontwikkelingen die relevant zijn voor de eigen systemen en gegevens.
Beoordeel gevolgen van automatisering, cloudgebruik en afhankelijkheid van leveranciers.
Werk scenario's bij en houd herstel- en responsplannen actueel.
Investeer in samenwerking en vaardigheden naast technische hulpmiddelen.
6. Adaptieve beveiligingsarchitectuur
Ontwerp beveiliging rond bedrijfsdoelen, gegevens en dreigingen. Pas architectuur aan wanneer systemen, leveranciers of risico's veranderen.
Ken kritieke activa, gegevensstromen en afhankelijkheden.
Beperk toegang tot wat nodig is; controleer identiteit en context passend bij het risico.
Gebruik gelaagde maatregelen, zoals segmentatie, logging en herstelvoorzieningen.
Beoordeel cloudverantwoordelijkheden samen met de gekozen dienst en configuratie.
5. Naleving als basis, niet als doel
Wet- en regelgeving en interne normen geven minimumverwachtingen. Een vinkje op een controlelijst bewijst niet vanzelf dat een organisatie bestand is tegen haar concrete dreigingen.
Breng toepasselijke verplichtingen en verantwoordelijken in kaart.
Vertaal vereisten naar werkende controles en bewijsstukken.
Beoordeel resterend risico en uitzonderingen periodiek.
Richt verbeteringen ook op risico's die niet in een formele controlelijst staan.