PENETRATIONSTESTER I RELATION TILL EU:S CYBERSÄKERHETSKRAV

Högskoleingenjörsutbildning i IT-Ingenjör
Cybersäkerhet och digital infrastruktur

Beruk Gebru
Raman Rostam
Marc Söder

HÖGSKOLAN I BORÅS

Program: IT-ingenjör – Cybersäkerhet och digital infrastruktur
Svensk titel: Penetrationstester i relation till EU:s cybersäkerhetskrav
Engelsk titel: Penetration testing in relation to EU's cybersecurity regulations
Utgivningsår: 2025
Författare: Beruk Gebru, Raman Rostam & Marc Söder
Handledare: Håkan Alm
Examinator: Carl Tinnsten
Nyckelord: CIA, CRA, Cybersecurity, DREAD, NIS2, Penetrationstestning, RED DA

Sammanfattning (svenska)

I takt med den ökande digitaliseringen och spridningen av IoT- och OT-enheter har cybersäkerhet blivit en allt viktigare fråga. För att möta den växande hotbilden har EU infört nya direktiv, såsom NIS2, Cyber Resilience Act (CRA) och den delegerade akten inom Radio Equipment Directive (RED-DA), vilka ställer skärpta krav på säkerhet i digitala produkter genom hela deras livscykel.

Syftet med denna studie är att undersöka i vilken utsträckning nätverksanslutna enheter uppfyller dessa säkerhetskrav. Genom praktiska penetrationstester av sju utvalda enheter, analyserades deras motståndskraft mot cyberhot. Testningen genomfördes i en kontrollerad labbmiljö med hjälp av verktyg som Nmap, Hydra, Burp Suite och OWASP ZAP, och resultaten analyserades med stöd av DREAD-modellen.

Resultaten visar att även om vissa enheter hade goda grundläggande skydd, till exempel genom att minimera antalet öppna portar, uppvisade flera enheter allvarliga brister, särskilt kopplat till DoS-skydd och autentiseringsmekanismer. Dessa sårbarheter strider mot centrala delar av de aktuella EU-direktiven. Slutsatsen är att flera av de testade enheterna inte uppfyller kraven fullt ut, vilket understryker behovet av förbättrade säkerhetsåtgärder för att uppnå efterlevnad och minska risken för framtida attacker.

Abstract (Engelska)

As digitalization and the spread of IoT and OT devices continue to grow, cybersecurity has become an increasingly important issue. In response to the growing threat landscape, the EU has introduced new directives, such as NIS2, the Cyber Resilience Act (CRA), and the delegated act within the Radio Equipment Directive (RED-DA), which impose stricter security requirements on digital products throughout their lifecycle.

The purpose of this study is to investigate the extent to which network-connected devices meet these security requirements. Through practical penetration testing of seven selected devices, their resilience to cyber threats was analyzed. The testing was conducted in a controlled laboratory environment using tools such as Nmap, Hydra, Burp Suite, and OWASP ZAP, and the results were analyzed with the support of the DREAD model.

The results show that while some devices had strong basic protection, such as minimizing the number of open ports, several devices exhibited serious vulnerabilities, particularly related to DoS protection and authentication mechanisms. These vulnerabilities violate key aspects of the current EU directives. The conclusion is that many of the tested devices do not fully meet the requirements, highlighting the need for improved security measures to achieve compliance and reduce the risk of future attacks.

ORDLISTA

NIS2: Network and Information Security Directive 2 – EU-direktiv för stärkt cybersäkerhet inom medlemsländerna.

NIST: National Institute of Standards and Technology – Amerikansk myndighet som publicerar ramverk och standarder för cybersäkerhet

CRA: Cyber Resilience Act – EU-förordning som ställer cybersäkerhetskrav på digitala produkter och tjänster

RED DA: Radio Equipment Directive Delegated Act – EU-tillägg till RED-direktivet med krav på cybersäkerhet i radioutrustning.

CIA: Confidentiality, Integrity, Availability – Grundprinciper inom informationssäkerhet.

IoT: Internet of Things – Nätverksanslutna fysiska enheter som samlar och utbyter data.

OT: Operational Technology – Gränssnittet mellan dator och industriella maskiner

IP-Adress: Internet Protocol-adress – En unik identifierare för en enhet i ett nätverk.

DNS: Domain Name System – Översätter domännamn till IP-adresser.

HTTP/S: Hypertext Transfer Protocol (Secure) – Protokoll för kommunikation över webben, med HTTPS som krypterad version

SSH: Secure Shell – Protokoll för säker fjärranslutning till system

FTP: File Transfer Protocol – Protokoll för filöverföring mellan datorer.

ICMP: Internet Control Message Protocol – Används för felsökning och kontroll av nätverksanslutningar (t.ex. ping)

TCP: Transmission Control Protocol – Pålitligt transportprotokoll i nätverk

UDP: User Datagram Protocol – Transportprotokoll med låg latens men utan garantier för leverans

CVE: Common Vulnerabilities and Exposures – Officiellt katalogiserade sårbarheter i mjukvara och hårdvara

Innehållsförteckning

1. INLEDNING

1.1 Bakgrund

I takt med den snabba digitaliseringen har cybersäkerhet blivit en allt viktigare fråga för företag. Förr handlade cybersäkerhet främst om att skydda enskilda datorer och nätverk från virus och intrång. Idag, med komplexa IT-miljöer, molntjänster och ökade cyberhot, krävs mer avancerade säkerhetsåtgärder. Denna utveckling har lett till ett ökat behov av att förstå och hantera risker relaterade till cybersäkerhet på ett systematiskt sätt.

Risk inom cybersäkerhet definieras som sannolikheten att ett hot utnyttjar en sårbarhet och orsakar negativa konsekvenser för en organisation eller ett system (NIST, 2012). Risk är en funktion av tre centrala faktorer: hot, sårbarhet och konsekvens. Ett hot ("threat") utgörs av en potentiell händelse eller aktör med avsikt att orsaka skada, exempelvis genom cyberattacker, skadlig kod eller insiderhot. Sårbarhet ("vulnerability") refererar till en brist i ett system, en applikation eller en process som kan utnyttjas för att genomföra en attack. När en angripare identifierar och utnyttjar en sådan sårbarhet genom en exploit, en metod eller kod som möjliggör intrång, kan detta leda till allvarliga säkerhetsincidenter.

En central aspekt vid riskbedömning är attackytan ("attack surface"), vilket avser alla möjliga ingångspunkter där en angripare kan försöka få åtkomst till ett system. Attackytan kan inkludera nätverksprotokoll, applikationer, enheter och mänskliga faktorer, såsom social engineering. Genom att minska attackytan, exempelvis genom segmentering av nätverk eller principer för minsta möjliga behörighet, kan organisationer begränsa potentiella angreppsvektorer. Cybersäkerhet handlar därför inte enbart om tekniska skyddsåtgärder utan även om att förstå och hantera risker kopplade till hot, sårbarheter och attackytor.

Den snabba utvecklingen av IoT (Internet of Things) och OT (Operational Technology) har förvandlat industrier, smart infrastruktur och konsumentekosystem, men samtidigt introducerat omfattande cybersäkerhetsutmaningar. De inneboende sårbarheterna hos sammanlänkade enheter innebär ökade risker, särskilt i kritiska sektorer såsom hälso- och sjukvård samt industriell automation. IoT-systemens resursbegränsade natur i kombination med bristande säkerhet-by-design-principer förvärrar hoten mot dataintegritet, integritetsskydd och operativ säkerhet (Chiara 2022).

Den växande hotbilden mot europeiska organisationer förstärker behovet av gemensamma cybersäkerhetsramverk. Statligt sponsrade aktörer, organiserade cyberkriminella nätverk och avancerade hotaktörer riktar sig allt oftare mot samhällskritiska funktioner, särskilt i energisektorn, tillverkningsindustrin och sjukvården. Dessa aktörer använder sig av sofistikerade metoder, som supply chain-attacker, ransomware och utnyttjande av zero-day sårbarheter, vilket gör traditionella försvarsmekanismer otillräckliga. Enligt ENISA Threat Landscape 2023 ökade antalet riktade attacker mot europeiska organisationer kraftigt under året, med särskild betoning på attacker mot offentliga institutioner och leverantörskedjor (European Union Agency for Cybersecurity, 2023).

Ett av de mest uppmärksammade exemplen på cyberattacker i Europa är WannaCry-ransomwareprogram som spreds globalt i maj 2017. Attacken utnyttjade en sårbarhet i Windows-operativsystemet via EternalBlue som utvecklades av den amerikanska underrättelsetjänsten NSA (National Security Agency), vilket möjliggjorde snabb spridning inom och mellan nätverk. Särskilt hårt drabbades den brittiska hälso- och sjukvården, NHS

(National Health Service), där mer än 200 000 datorer påverkades, vilket ledde till inställda operationer, avbokade läkarbesök och nedstängda IT-system i hela landet. Attacken avslöjade allvarliga brister i uppdateringsrutiner och segmentering inom kritisk infrastruktur. WannaCry blev en väckarklocka för många länder och organisationer inom EU och visade tydligt behovet av både tekniska skydd och strukturerad cybersäkerhetsstyrning (Europol, 2018)

Europeiska unionen har intensifierat sitt arbete med att utveckla lagstiftning som stärker säkerheten i digitala produkter och tjänster. Initiativ såsom CRA (Cyber Resilience Act) och den delegerade akten inom RED DA (Radio Equipment Directive) syftar till att skapa en mer robust och enhetlig säkerhetsstandard inom EU. Dessa regelverk kompletterar tidigare direktiv, såsom NIS 2, och utgör en del av en bredare strategi för att säkerställa cybersäkerhet i en alltmer uppkopplad värld. Med den ökande hotbilden har behovet av enhetliga cybersäkerhetsregler blivit alltmer påtagligt, och dessa lagstiftningsinitiativ syftar till att skydda både företag och konsumenter från cyberattacker.

Den föreslagna CRA av Europeiska unionen representerar en avgörande utveckling, i en tid präglad av eskalerande cyberhot och den genomgripande integrationen av digital teknik, som syftar till att åtgärda kritiska luckor i säkerheten för hård- och mjukvaruprodukter samtidigt som den samspelar med befintliga regleringsinstrument. Eftersom cyberattacker alltmer utnyttjar sårbarheter i sammankopplade system, från konsumentenheter till industriell infrastruktur, försöker CRA:s horisontella tillvägagångssätt att etablera grundläggande cybersäkerhetsstandarder för alla produkter med digitala element. Detta initiativ understryker inte bara EU:s åtagande att skydda sin digitala inre marknad utan väcker också viktiga frågor om dess samspel med sektorspecifika regleringar och bredare cybersäkerhetsstrategier, särskilt det reviderade direktivet om nät- och informationssäkerhet (NIS 2). Att förstå detta samspel är viktigt för att utvärdera CRA:s potential att förbättra systemisk motståndskraft samtidigt som man undviker regulatorisk fragmentering (Eckhardt & Kotovskaia 2023).

För att stärka cybersäkerheten och skydda kritiska verksamheter har EU infört flera viktiga direktiv. NIS-direktivet (Network and Information Security Directive), som först antogs 2016, har utformats för att säkerställa att samhällsviktiga tjänster har en grundläggande skyddsnivå. Det förnyade NIS2-direktivet, som trädde i kraft 2023, ställer högre krav på organisationer, inklusive utvidgade rapporteringsskyldigheter vid incidenter och ökat ansvar för företagsledningar (Singh 2023). Samtidigt reglerar RED DA cybersäkerhetsaspekterna för trådlösa enheter, inklusive IoT-produkter, och ställer krav på inbyggda säkerhetsfunktioner för att minska riskerna för obehörig åtkomst och manipulation (Falch, Henten, Kaloudis & Windekiælde 2024).

Företag och organisationer ställs därför inför allt högre säkerhetskrav, vilket innebär att de inte bara behöver ha grundläggande skydd utan även genomföra riskanalyser, penetrationstester och ha tydliga rutiner för incidenthantering. Detta markerar en förändring från ett tidigare reaktivt säkerhetstänk till ett mer proaktivt förhållningssätt, där cybersäkerhet integreras i hela livscykeln för digitala produkter och tjänster. För att möta dessa krav krävs inte bara investeringar i teknik utan även utveckling av intern kompetens, kontinuerlig utbildning av personal och samarbete med externa säkerhetsexperter. I praktiken innebär detta att säkerhetsaspekter måste beaktas redan i designfasen av system och applikationer, enligt principen security by design. Dessutom blir förmågan att snabbt upptäcka, begränsa och återhämta sig från incidenter avgörande för att upprätthålla verksamhetskontinuitet (ISO 27001).

1.2 Problemformulering

Med de skärpta EU:s cybersäkerhetskrav genom direktiven NIS2, RED DA och CRA, uppstår ett behov av att bedöma i vilken utsträckning aktuella enheter uppfyller dessa krav i praktiken. Särskilt viktigt är att utvärdera säkerheten hos enheter som är utsatta för nätverksbaserade hot och att identifiera potentiella sårbarheter innan de kan utnyttjas.

1.3 Syfte och forskningsfrågor

Syftet med denna undersökning är att utvärdera huruvida de enheter som testas uppfyller de säkerhetskrav som fastställs i de nya EU-direktiven, såsom NIS2, RED DA och CRA. Dessa direktiv syftar till att stärka cybersäkerheten och minska riskerna för sårbarheter i nätverksanslutna enheter.

- Hur väl uppfyller de testade enheterna säkerhetskraven enligt NIS2-, RED DA- och CRA-direktiven?
- Vilka sårbarheter kan identifieras genom penetrationstester av de utvalda enheterna?
- Vilken inverkan har de identifierade sårbarheterna på enheternas förmåga att uppfylla EU:s cybersäkerhetskrav?

För att besvara dessa frågor kommer studien att kombinera en teoretisk genomgång av relevanta regelverk och säkerhetsstandarder med praktiska penetrationstester. Genom att utföra penetrationstester på de valda enheterna kan vi identifiera potentiella sårbarheter och analysera om dessa utgör en risk i förhållande till de krav som ställs i EU-direktiven.

1.4 Avgränsningar

Denna studie genomfördes inom ett begränsat tidsfönster, vilket påverkade både förberedelsefasen och genomförandet av penetrationstesterna. Enheterna som testades var förutbestämda av företaget som studien genomfördes i samarbete med, detta innebar att testverktyg och metodik för penetrationstester fick anpassas efter de specifika enheterna och den tid som fanns tillgänglig.

Vidare påverkades testningens omfattning av den tid som fanns för att genomföra de praktiska experimenten. Detta ledde till att vissa testprocedurer, såsom djuplodande analyser eller upprepade tester för att säkerställa resultatens tillförlitlighet, inte kunde genomföras på alla enheter. Därmed är resultaten från testningen beroende av de specifika förhållandena under testperioden och kan inte helt generaliseras till alla typer av nätverksanslutna enheter som omfattas av EU:s direktiv.

Trots dessa tidsmässiga begränsningar strävar denna studie efter att ge en tillräcklig inblick i hur väl de testade enheterna uppfyller de krav som ställs av NIS2, RED DA och CRA, men resultaten bör betraktas inom ramen för de givna förutsättningarna.

2. LITTERATURGENOMGÅNG / TEORI

2.1 Cybersäkerhet

Cybersäkerhet är idag mycket viktigt för både privatpersoner och företag, särskilt eftersom allt fler människor använder teknik i sin vardag. Om det saknas säkerhetsåtgärder kan det få allvarliga konsekvenser, som att bli utsatt för cyberattacker. Exempelvis kan en DoS (Denial of Service)-attack riktad mot ett företag leda till att dess webbplatser och system helt slås ut. Kraven på cybersäkerhet har ökat avsevärt under de senaste åren. Ett tydligt exempel är Colonial Pipeline- och SolarWinds-attackerna i USA år 2021, vilka avslöjade stora brister i regeringens beredskap för cyberattacker. Dessa incidenter orsakade betydande skador på landets ekonomiska säkerhet och har varit en drivkraft bakom skärpta krav på cybersäkerhet (Snider, Shandler, Zandani & Canetti 2021).

2.1.1 Internet Of Things (IoT)

Utvecklingen av IoT har revolutionerat både konsumentmarknaden och industrin genom att koppla samman fysiska enheter med internet. Det möjliggör automatisering, fjärrstyrning och realtidsövervakning inom allt från smarta hem till sjukvård och tillverkningsindustri. I industriella miljöer har denna utveckling också suddat ut gränsen mellan traditionella IT-system och OT, det vill säga de system som styr fysiska processer som exempelvis produktionslinjer, vattenrening eller energidistribution.

Trots dess fördelar har IoT och OT skapat en komplex och snabbt växande attackyta. Många enheter har begränsade resurser vad gäller minne och processorkraft, vilket gör att de ofta saknar robusta säkerhetsfunktioner. Dessutom används kommunikationsprotokoll som MQTT, Modbus och OPC UA, protokoll som från början inte designades med säkerhet i åtanke. MQTT till exempel, ett lättviktigt protokoll som ofta används för att överföra meddelanden mellan IoT-enheter, saknar inbyggda mekanismer för kryptering och autentisering om det inte konfigureras korrekt. I OT-miljöer förekommer även att system körs med föråldrade operativsystem och sällan uppdateras, vilket ytterligare ökar risken (Bhole, Kastner & Sauter 2024).

En stor del av säkerhetsproblemen beror på svagheter i tillverkningsfasen, exempelvis bristfälliga uppdateringsrutiner, standardlösenord och otillräcklig kryptering. Dessa sårbarheter utnyttjas ofta av angripare som en inkörsport till större nätverk. Ett känt exempel är Mirai-botnettet, som 2016 infekterade tiotusentals IoT-enheter och användes för att genomföra en massiv DDoS (Distributed Denial of Service)-attack, en överbelastningsattack där ett flertal enheter skickar stor mängd trafik till ett system, mot DNS-leverantören Dyn, vilket slog ut stora delar av internet i Nordamerika och Europa (Antonakakis, April, Bailey, Bernhard, Bursztein, Cochran, Halderman, Invernizzi, Kumar, Lever, Ma, Mason, Menscher, Sullivan, Thomas & Zhou 2017).

Med tanke på hur IoT och OT nu ofta integreras i kritisk infrastruktur är cybersäkerheten i dessa system avgörande. För att hantera riskerna krävs det att både tillverkare, systemintegratörer och användare implementerar åtgärder som segmentering av nätverk, autentisering, krypterad kommunikation, kontinuerlig övervakning samt regelbundna programvaruuppdateringar. Denna komplexitet och sårbarhet har även uppmärksammats på regulatorisk nivå, där exempelvis EU:s cybersäkerhetsdirektiv som NIS2 och CRA ställer specifika krav på hur uppkopplade enheter, inklusive de som används i OT-miljöer, ska säkras genom hela deras livscykel.

2.1.2 Brute-Force-Attacker

En brute-force-attack är en metod där en angripare systematiskt prövar alla möjliga kombinationer av användarnamn och lösenord tills rätt information hittas. Detta är en av de äldsta men fortfarande vanligaste attackmetoderna mot autentiseringssystem, särskilt sådana utan begränsningar för inloggningsförsök eller svaga lösenordspolicies. Brute-force-attacker kan automatiseras med verktyg som Hydra eller Medusa, vilket gör det möjligt att snabbt testa tusentals kombinationer. Angreppets effektivitet ökar ytterligare om angriparen använder en så kallad lösenordslista, det vill säga en samling av vanliga eller tidigare läckta lösenord.

Bristande skydd mot denna typ av attack kan leda till obehörig åtkomst till känsliga system, vilket i sin tur kan resultera i dataläckor, systemkompromettering eller andra allvarliga säkerhetsincidenter. För att förhindra brute-force-attacker rekommenderas det att implementera säkerhetsåtgärder såsom kontolåsning efter ett visst antal misslyckade försök, CAPTCHA-lösningar, multifaktorautentisering, samt IP-blockering. Dock har angripare utvecklat motstrategier, exempelvis genom så kallad IP-hopping, där de kontinuerligt byter IP-adress under attacken för att undgå blockering och förhindra att attacken upptäcks. Detta gör det svårare för försvarssystem att identifiera och stoppa pågående brute-force-attacker, vilket ytterligare understryker behovet av flerlagriga försvarsåtgärder. (Vugdelija, Nedeljković, Kojić, Lukić & Vesić 2021).

2.1.3 Denial Of Service

En DoS-attack är en form av cyberattack som syftar till att störa tillgängligheten hos ett system, en tjänst eller ett nätverk genom att överbelasta dess resurser. Detta uppnås vanligtvis genom att en angripare skickar en stor mängd förfrågningar eller datapaket på mycket kort tid, vilket medför att systemet inte längre kan hantera legitim trafik och därmed blir otillgängligt för användare.

Det finns flera olika metoder för att genomföra DoS-attacker, däribland olika typer av så kallade flood-attacker. En vanlig metod är ICMP flood, även känd som ping flood, där angriparen skickar en stor mängd ICMP Echo Request-paket (ping) till målenheten. Om målet försöker svara på varje förfrågan kan dess bandbredd och processorkraft snabbt förbrukas, vilket leder till avbrott i tjänsten (Cloudflare u.å.).

En annan typ av attack är SYN flood, som utnyttjar TCP:s trevägshandskakning. I denna attack skickar angriparen en stor mängd TCP SYN-paket till servern, men besvarar inte serverns SYN-ACK-svar. Detta lämnar servern med ett stort antal halvöppna anslutningar, vilket blockerar resurser och förhindrar nya, legitima anslutningar från att etableras. På detta sätt kan angriparen effektivt få servern att upphöra att fungera korrekt (Schuba, Krsul, Kuhn, Spafford, Sundaram & Zamboni, 1997).

2.1.4 Common Vulnerabilities and Exposures (CVE)

CVE (Common Vulnerabilities and Exposures) är en systematisk databas som innehåller identifierade säkerhetsbrister i mjukvara och hårdvara. Den används globalt av säkerhetsexperter och IT-organisationer för att dela och spåra information om säkerhetshot. Varje sårbarhet tilldelas ett unikt CVE-ID, vilket gör det lättare att identifiera och diskutera säkerhetsproblem. Målet är att skapa en gemensam referenspunkt för att kunna kommunicera om sårbarheter utan att behöva förlita sig på olika benämningar och terminologier. CVE gör det möjligt att snabbt identifiera och åtgärda kända sårbarheter, vilket är avgörande för att minska risken för attacker som utnyttjar dessa brister (CVE u.å).

När en säkerhetsbrist upptäcks och verifieras av en säkerhetsexpert eller ett företag, ansöker de om ett CVE-ID genom en av de officiella CVE-programmen (till exempel MITRE eller National Vulnerability Database). CVE-databasen innehåller detaljerad information om varje sårbarhet, inklusive dess allvarlighetsgrad (baserat på CVSS (Common Vulnerability Scoring System)-skalor), drabbade system, och tillgängliga uppdateringar eller lösningar. Denna information gör det möjligt för säkerhetsteam att snabbt vidta åtgärder för att åtgärda de sårbarheter som har identifierats (CVE, u.å).

CVE-databasen spelar en central roll i att hantera riskerna för företag och organisationer. Genom att hålla sig uppdaterade om CVE-listan och de sårbarheter som har identifierats, kan företag proaktivt uppdatera sina system och programvara för att förhindra attacker. Detta är särskilt viktigt när det gäller kritisk infrastruktur och IoT-enheter, där sårbarheter kan utnyttjas för att utföra massiva attacker. I IoT- och OT-miljöer, där många enheter körs med föråldrad programvara eller bristfällig säkerhet, kan CVE-databasen ge en vägledning för att identifiera risker och sårbarheter. Eftersom många IoT-enheter inte har möjlighet att implementera avancerade säkerhetsåtgärder är det avgörande att hålla dessa enheter uppdaterade med de senaste säkerhetsuppdateringarna och CVE-patcharna. Ett exempel är CVE-2017-5523, som var en säkerhetsbrist i flera äldre IoT-enheter som användes för att utföra DDoS-attacker.

2.2 Network and Information Security Directive 2

Ett av de viktigaste direktiven inom EU som företag måste följa är NIS2. Det syftar till att stärka cybersäkerheten genom högre krav på riskhantering, incidentrapportering och skydd av nätverk och system. NIS2-direktivet är ett centralt lagstiftningsinstrument inom EU:s cybersäkerhetsstrategi och representerar en betydande uppdatering av det tidigare NIS-direktivet från 2016. Syftet med NIS2 är att stärka den gemensamma säkerhetsnivån inom unionen, genom att införa striktare krav på cybersäkerhet för offentliga och privata aktörer vars verksamhet anses vara av väsentlig betydelse för samhällets funktion (Vandezande, 2024).

En av de mest framträdande förändringarna i NIS2 är utvidgningen av tillämpningsområdet. Direktivet omfattar nu ett större antal sektorer, inklusive energi, transport, hälso- och sjukvård, digital infrastruktur, offentliga förvaltningar och tillverkande industri. Företag och organisationer som faller inom dessa kategorier måste följa de nya reglerna, vilket innebär att de måste ha en dokumenterad cybersäkerhetsstrategi, kontinuerligt genomföra riskbedömningar och säkerställa att skyddet av nätverks- och informationssystem är i linje med direktivets krav (Vandezande, 2024).

En annan viktig del av NIS2 är de ökade kraven på incidentrapportering. Organisationer förpliktas att snabbt rapportera allvarliga säkerhetsincidenter till relevanta nationella myndigheter inom en viss tidsram, oftast inom 24 timmar från upptäckt, för att möjliggöra ett effektivt och koordinerat svar. Detta är en väsentlig förändring jämfört med det tidigare direktivet, där kraven på rapportering var mindre tydligt definierade (Vandezande, 2024).

NIS2 ställer även krav på att organisationer implementerar förebyggande tekniska och organisatoriska åtgärder, vilket inkluderar användning av säkerhetsramverk, kryptering, nätverksövervakning, autentisering och åtkomstkontroller. Dessa åtgärder ska inte bara minska sannolikheten för incidenter, utan även begränsa konsekvenserna om en incident skulle inträffa. Här uppmuntras också användningen av penetrationstester och sårbarhetsanalyser som en del av det proaktiva arbetet med cybersäkerhet (Singh, 2023).

2.3 Cyber Resilience Act

Ett annat direktiv är cyber resilience act, istället att rikta sig mot företag som NIS2 så riktar CRA direktivet sig mot mjuk- och hårdvara. Då cyberattacker är allt vanligare så har man inom EU bestämt att man ska ställa högre krav på produktutvecklare för att produkterna de säljer är säkra, detta genom att säkerställa att sårbarheter hanteras proaktivt genom hela processen från start till slut.

Målet med lagstiftningen är att säkerställa att digitala produkter som säljs inom EU är utformade med cybersäkerhet som en integrerad del från början, det vill säga att säkerhet ska beaktas redan i design- och utvecklingsfasen, inte som ett tillägg i efterhand. Utöver detta ställs krav på att leverantörer kontinuerligt ska hantera sårbarheter, inklusive att kunna upptäcka, rapportera och åtgärda dessa även efter att produkten har släppts på marknaden. Det innebär att organisationer behöver etablera robusta rutiner för exempelvis sårbarhetshantering, regelbundna säkerhetsuppdateringar och långsiktig support (Jara, Martinez & Sanchez 2024).

CRA kräver också att tillverkare och leverantörer genomför systematiska riskbedömningar och dokumenterar säkerhetsrelaterade åtgärder för varje produkt. Denna dokumentation ska finnas tillgänglig för tillsynsmyndigheter och kunna användas som grund vid en eventuell granskning. Användare ska dessutom ges tydlig information om produktens säkerhetsfunktioner, underhållsbehov och potentiella cybersäkerhetsrisker.

Lagen trädde i kraft den 10 december 2024, och kommer att börja tillämpas den 11 december 2027. Under denna övergångsperiod ges aktörer inom teknik- och produktsektorn möjlighet att anpassa sina processer, system och dokumentation för att uppfylla de nya kraven. När CRA börjar tillämpas kommer produkter som inte uppfyller regelverket inte längre att tillåtas på den europeiska marknaden.

Genom att införa gemensamma säkerhetsstandarder för digitala produkter syftar CRA till att förbättra det digitala ekosystemets motståndskraft mot cyberattacker och att främja förtroendet för uppkopplade teknologier inom EU. I samspel med andra regelverk, såsom NIS2 och RED DA, utgör CRA en central del av Europeiska unionens strategi för att skapa en mer säker och robust digital infrastruktur (European Cyber Resiliant Act u.å.).

2.4 Radio Equipment Directive Delegation Act

I *Direktiv 2014/53/EU om harmonisering av medlemsstaternas lagstiftning om tillhandahållande av radioutrustning på marknaden* så kompletterar det befintliga RED (Radio Equipment Directive) och blir bindande från och med den 1 augusti 2025. RED, som först antogs 2014, syftar till att harmonisera reglerna för marknadsföring av radioutrustning inom EU och fastställer tekniska krav för tillverkare av trådlösa produkter, såsom routrar, mobiltelefoner och IoT-enheter.

Denna kompletterande rättsakt, *Kommissionens delegerade förordning (EU) 2022/30 av den 29 oktober 2021 om komplettering av direktiv 2014/53/EU vad gäller tillämpningen av de väsentliga krav som avses i artikel 3.3 d, e och f i det direktivet (Text av betydelse för EES)*, inför särskilda cybersäkerhetskrav för vissa kategorier av radioutrustning för att stärka skyddet av nätverk, personuppgifter och mot bedrägeri er. En bakgrund till behovet av sådana regler är tidigare incidenter, såsom Mirai-botnet, där oskyddade IoT-enheter kompromettterades och användes för att genomföra omfattande DDoS-attacker.

De specifika kraven som träder i kraft genom förordningen avser artikel 3.3 punkterna d, e och f i RED-direktivet:

- Punkt d föreskriver att utrustningen ska vara konstruerad för att inte skada nätverket, dess funktion eller orsaka oacceptabel försämring av tjänsten.
- Punkt e kräver att utrustning som behandlar person- eller lokaliseringsdata ska ha inbyggda funktioner för att säkerställa dataskydd och integritet.
- Punkt f avser skydd mot bedrägerier vid överföring av penningvärden eller virtuella valutor via radioutrustning.

Dessa tillägg är avsedda att säkerställa att trådlös utrustning som släpps ut på den inre marknaden inte medför säkerhetsbrister som kan utnyttjas av illvilliga aktörer.

2.5 CIA-Triangeln Och Dess Tillämpningar

Inom cybersäkerhet utgör CIA-triangeln (Confidentiality, Integrity, Availability) ett centralt ramverk för att skydda system och data. Den syftar till att säkerställa att information endast är tillgänglig för behöriga (konfidentialitet), att den är korrekt och oförändrad (integritet), samt att den är åtkomlig för legitima användare (tillgänglighet) (Covert, Francis, Steinhagen & Streff 2020).

NIST (National Institute of Standards and Technology) tillämpar denna modell i sitt säkerhetsramverk, där fokus ligger på praktiska och tekniska åtgärder för att motverka hot såsom brute-force-attacker. Exempel på sådana åtgärder inkluderar begränsning av inloggningsförsök, stark lösenordshantering samt övervakning och loggning av misstänkt aktivitet. Dessa insatser stärker systemets motståndskraft och minskar risken för obehörig åtkomst (Covert et al. 2020).

CIA-triangeln återfinns även som en underliggande princip i flera EU-direktiv, såsom NIS2, RED DA och CRA. Dessa regelverk syftar till att höja den digitala säkerheten inom EU genom att kräva att produkter och tjänster uppfyller grundläggande cybersäkerhetskrav. Till exempel betonar NIS2 vikten av incidenthantering och riskanalys kopplat till tillgänglighet och integritet, medan RED DA och CRA ställer krav på att nätverksanslutna produkter ska skyddas mot obehörig åtkomst och dataläckage, vilket direkt relaterar till konfidentialitet och integritet. Genom att implementera åtgärder enligt CIA-modellen kan organisationer bättre uppfylla dessa direktiv och stärka sin totala cybersäkerhet.

2.6 Penetrationstestning

Penetrationstestning är en metod för att identifiera och utnyttja sårbarheter i system, nätverk eller applikationer med målet att förbättra säkerheten. Processen börjar med informationsinsamling, vilket ofta görs genom Vulnerability Assessment. Vulnerability Assessment innebär att man skannar och analyserar system och nätverk för att identifiera potentiella sårbarheter, exempelvis genom verktyg som Nmap (Network Mapper), Nessus eller OpenVAS. Denna fas hjälper testaren att förstå var svagheterna finns.

När tillräckligt med information har samlats in genomförs själva penetrationstestet. Testaren försöker då aktivt utnyttja de identifierade sårbarheterna för att bedöma hur en angripare skulle kunna få obehörig åtkomst eller orsaka skada. Det finns tre huvudsakliga metoder för penetrationstestning: black box, grey box och white box. Vid black box-testning har testaren ingen förkunskap om nätverket eller systemet som testas. Grey box-testning innebär att testaren har viss förkunskap om systemet men saknar full insyn i dess uppbyggnad. White box-testning

innebär att testaren har fullständig kunskap om systemets och nätverkets struktur, och denna typ av testning genomförs oftast internt inom en organisation. (Goel & Mehtre 2015).

2.6.1 Cyber kill chain modellen

Cyber Kill Chain-modellen, utvecklad av Lockheed Martin, beskriver de olika faserna av en cyberattack och används för att identifiera och motverka hot i ett tidigt skede. Modellen består av sju steg: rekognosering, beväpning, leverans, exploatering, installation, kommando- och kontrollkommunikation samt måluppfyllelse. Genom att analysera dessa faser kan organisationer implementera effektiva försvarsmekanismer för att bryta angriparens kedja och minimera potentiella skador.

Inom penetrationstestning fungerar Cyber Kill Chain som en metodologisk ram för att simulera verkliga angreppsmetoder och identifiera sårbarheter i IT-miljöer. Genom att följa attackkedjans struktur kan etiska hackare systematiskt testa en organisations säkerhetsmekanismer och utvärdera dess förmåga att detektera och stoppa angrepp i olika faser. Särskilt relevant är testning av initial åtkomst, lateral rörelse och uthållighet inom system, vilket ger insikter i hur väl försvarsåtgärder såsom intrångsdetektering och incidentrespons fungerar i praktiken. Kombinationen av Cyber Kill Chain och penetrationstestning möjliggör således en proaktiv säkerhetsstrategi där sårbarheter kan åtgärdas innan de utnyttjas av illasinnade aktörer (Lockheed Martin u.å).

2.6.2 MITRE ATT&CK Framework

MITRE ATT&CK (Adversarial Tactics, Techniques, and Common Knowledge) är en omfattande databas över verkliga angreppsmetoder som används av hotaktörer. Till skillnad från Cyber Kill Chain, som fokuserar på en sekventiell angreppsmodell, erbjuder ATT&CK en mer detaljerad och icke-linjär representation av attacker genom att kategorisera dem i taktiker och tekniker baserade på faktisk hotinformation. Ramverket används inom cybersäkerhet för att kartlägga intrångsaktiviteter, utveckla försvarsåtgärder och förbättra incidentrespons.

I relation till penetrationstestning fungerar MITRE ATT&CK som ett strukturerat ramverk för att simulera verkliga cyberangrepp och identifiera organisationers sårbarheter. Genom att kartlägga en attack mot ATT&CK:s klassifikation kan säkerhetsexperter förstå hur en angripare kan utnyttja specifika tekniker inom olika faser av intrånget. Detta komplimenterar Cyber Kill Chain-modellen genom att tillhandahålla en mer detaljerad och finfördelad analys av varje steg i attacken, särskilt avseende lateral rörelse, exfiltration och ihållande åtkomst. Kombinationen av Cyber Kill Chain, MITRE ATT&CK och penetrationstestning möjliggör en övergripande och offensiv säkerhetsstrategi där organisationer kan testa sin motståndskraft mot avancerade hot och proaktivt stärka sina skyddsmekanismer (MITRE u.å).

2.6.3 Penetrationstestnings verktyg

För att kunna utföra en effektiv penetrationstestning finns det ett antal verktyg som kan användas för olika faser av testprocessen, inklusive informationsinsamling, sårbarhetsidentifiering, exploatering och rapportering. Ett av de mest använda verktygen är Nmap, som främst används för nätverksskanning och portskanning. Nmap hjälper testaren att identifiera aktiva enheter i nätverket, identifiera öppna portar och bestämma vilka tjänster som körs på dessa portar. Denna information är avgörande för att förstå systemets attackyta och identifiera potentiella sårbarheter. Andra viktiga verktyg inom penetrationstestning inkluderar Metasploit Framework, Burp Suite, Wireshark, och Hydra, vilka används i olika faser av testningen (Nmap u.å).

Metasploit Framework är ett av de mest använda verktygen för exploatering. Det innehåller en omfattande samling av färdiga exploits och så kallade payloads (kod som körs efter att en sårbarhet har hittats, till exempel för att ta kontroll över ett system eller skicka ut information), som kan användas för att simulera attacker och testa sårbarheter i system. Metasploit möjliggör både manuell och automatiserad testning och används ofta för att verifiera om identifierade sårbarheter kan utnyttjas i praktiken (Raj & Walia 2020).

Burp Suite är ett kraftfullt verktyg för webbapplikationssäkerhet och används främst för att analysera och manipulera HTTP- och HTTPS-trafik. Det kan identifiera sårbarheter såsom SQL-injektion och cross-site scripting (XSS) genom automatiserade och manuella tester. Med hjälp av Burp Suite kan penetrationstestare kartlägga applikationers attackytor och analysera hur dataflöden kan manipuleras (Kim 2020).

Wireshark är ett nätverksanalysverktyg som används för att inspektera nätverkstrafik på nätverksnivå. Genom att fånga och analysera datapaket kan testare identifiera osäkra kommunikationsprotokoll, misstänkta nätverksaktiviteter och potentiella säkerhetsbrister i realtid. Detta gör Wireshark till ett ovärderligt verktyg vid både informationsinsamling och incidentanalys (Soepeno 2023).

Hydra är ett kraftfullt verktyg för brute-force-attacker mot autentiseringssystem. Det används för att testa lösenordssäkerhet genom att automatiskt försöka autentisera mot olika tjänster såsom SSH, FTP och webbapplikationer. Hydra hjälper säkerhetsexperter att identifiera svaga autentiseringsmetoder och rekommendera starkare säkerhetsåtgärder (Bakry, Adenan & Yussoff 2022).

Gobuster är ett open-source-verktyg som används inom cybersäkerhet för att identifiera dolda resurser på webbservrar, såsom underkataloger och filer, genom ordlistestyrda brute-force-attacker. Verktyget är särskilt vanligt inom penetrationstestning och bug bounty-program där säkerhetsgranskare söker efter potentiella sårbarheter (Oberheide 2018). Ett verktyg med liknande syfte är Wfuzz, som även det är open-source och används för så kallad webbfuzzing, en teknik där olika HTTP-parametrar automatiskt testas med hjälp av ordlistor för att identifiera sårbarheter och otillgängliga resurser (Wfuzz, u.å). Både Gobuster och Wfuzz utgör viktiga verktyg i en säkerhetsanalytikers arsenal för att kartlägga angripbara ytor i webbaserade applikationer.

Nikto är ett open-source-verktyg för webbserveranalys som används inom penetrationstestning för att identifiera potentiella säkerhetsbrister i webbapplikationer och webbtjänster. Verktyget genomför automatiserade tester mot webbservrar för att upptäcka kända sårbarheter, såsom föråldrad programvara, feltillstånd, samt konfigurationsproblem (CISA u.å).

Hping3 är ett kommandoverktyg som används för att göra nätverks- och penetrationstester. Hping3 har flera olika syften där den kan hantera saker som portscanning, firewall-bypass-testing, traceroute, floodattacker och spoofing. Programmet är bra för att stimulera DoS-attacker genom att skicka ofantligt stora ICMP-, UDP- eller TCP-paket.

Netcat, även känt som *nc*, är ett mångsidigt verktyg som använder TCP- och UDP-anslutningar för att läsa och skriva data över nätverk. Det fungerar på de flesta operativsystem och används både i offensiva och defensiva säkerhetssyften. Inom penetrationstestning kan Netcat till exempel användas för att skanna portar, överföra filer mellan system, skapa bakdörrar för fjärråtkomst samt testa kommunikationskanaler mellan klient och server. Dess flexibilitet gör det till ett ovärderligt verktyg för felsökning och nätverksanalys, och det beskrivs ofta som

"TCP/IP:s schweiziska armékniv" inom säkerhetsbranschen (GeeksforGeeks 2023, Vaishnavi 2024).

Utöver dessa verktyg spelar sårbarhetsskanning en viktig roll i penetrationstestning. Greenbone Vulnerability Management (tidigare OpenVAS) är ett avancerat verktyg för sårbarhetsskanning som identifierar säkerhetsbrister i nätverk och system genom kontinuerlig analys av kända sårbarheter. Verktyget använder en omfattande databas med CVE (Common Vulnerabilities and Exposures) för att identifiera potentiella hot och föreslå åtgärder (Greenbone Networks u.å).

För en mer specifik analys av kända sårbarheter används CVE-scanners**,** vilka hjälper testare att identifiera och verifiera säkerhetsbrister i system genom att jämföra dem med databaser som NVD (National Vulnerability Database). Genom att använda CVE-scanners kan säkerhetsexperter snabbt avgöra om ett system är sårbart för specifika attacker och därmed vidta nödvändiga åtgärder.

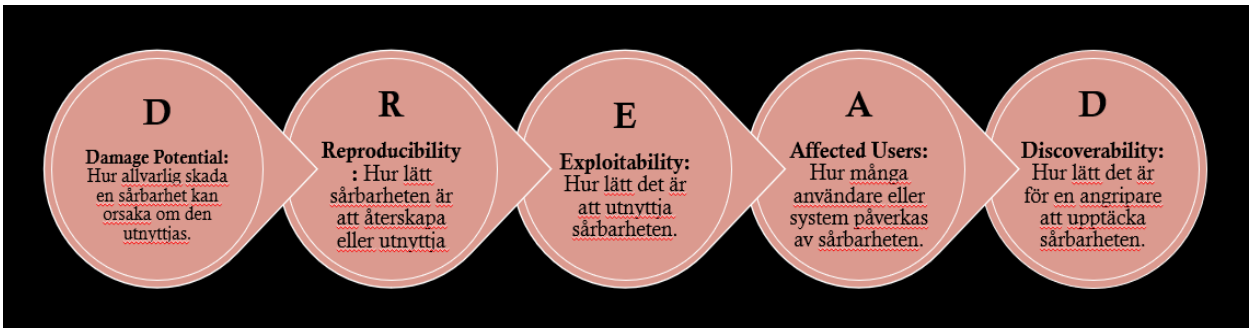
För att kunna utföra en effektiv penetrationstestning används en kombination av olika verktyg för informationsinsamling, sårbarhetsidentifiering, exploatering och rapportering. Ett av de mest använda operativsystemen för penetrationstestning är Kali Linux, en Linux-distribution specifikt utvecklad för säkerhetstestning. Kali Linux innehåller ett brett utbud av förinstallerade verktyg, bland annat Nmap, Metasploit, Burp Suite och Hydra, vilket gör det till en central plattform för både offensiv och defensiv cybersäkerhet.

Genom att kombinera dessa verktyg i en systematisk penetrationstestning kan organisationer identifiera och åtgärda säkerhetsbrister innan de exploateras av angripare. En effektiv testning kräver dock inte bara rätt verktyg, utan också en metodisk strategi som följer etablerade ramverk, såsom MITRE ATT&CK och Cyber Kill Chain för att säkerställa en omfattande analys av systemets säkerhet.

2.7 Hotmodeller

Hotmodeller, eller mer känt som threat modelling, är ett samlingsbegrepp för olika typer av metoder som används för att identifiera, analysera och förstå potentiella säkerhetshot mot system eller enheter. Syftet med dessa modeller är att på ett systematiskt sätt kartlägga hur ett system kan bli utsatt för attacker, vilka svagheter som finns, och hur dessa kan utnyttjas av angripare. Genom att använda hotmodeller kan man inte bara förutse möjliga attackvägar utan även implementera förebyggande åtgärder för att minska risken för att en attack sker. Hotmodellering bidrar också till en djupare förståelse för vilka typer av attacker systemet är mest sårbart för, samt vilka aktörer som sannolikt skulle kunna genomföra en attack (National Cyber Security Centre 2023).

DREAD (Damage Potential, Reproducibility, Exploitability, Affected Users och Discoverability) är en hotmodell som är till för att identifiera hot för att på bästa sätt kunna prioritera vilka hot som är viktigast (Se Figur 1). I DREAD-modellen rankas risker genom att tilldela en numerisk poäng till varje av de fem faktorerna. Skalan som vanligtvis används är en poäng från 0 till 10, där 0 representerar lägsta risk och 10 representerar högsta risk. Detta gör det möjligt att klassifiera hot och jämföra olika sårbarheter baserat på deras allvarlighetsgrad (Yuga, 2023).



Figur 1

STRIDE är en hotmodell som utvecklades år 1999 av Loren Kohnfelder och Praerit Garg vid Microsoft. Denna modell används för att klassificera säkerhetshot i sex distinkta kategorier: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service och Elevation of Privilege, vilket möjliggör en strukturerad och systematisk analys av potentiella sårbarheter i ett system. STRIDE bygger vidare på de tre grundläggande principerna i CIA-triaden, konfidentialitet, integritet och tillgänglighet, men utökar modellen med ytterligare tre säkerhetsaspekter: autentisering (authentication), auktorisation (authorization) och icke-förnekande (non-repudiation) (Kim, Kim & Kim 2022). De sammanlagt sex aspekterna motsvarar sex olika hotkategorier (se figur 2).

Threat	Desired Security Property
Spoofing	Authentication
Tampering	Integrity
Repudiation	Non-repudiation
Information Disclosure	Confidentiality
Denial of Service	Availability
Elevation of Privilege	Authorization

Figur 2

PASTA (Process for Attack Simulation and Threat Analysis) är en riskbaserad hotmodell som introducerades av Tony UcedaVélez och Marco M. Morana. Den skiljer sig från enklare modeller som STRIDE och DREAD genom att den fokuserar på att knyta säkerhetshot till affärsmål och riskhantering. PASTA består av sju steg: (1) Definition av mål, (2) Definition av teknisk omfattning, (3) Applikationsdekomposition och analys, (4) Hotanalys, (5) Sårbarhetsanalys, (6) Attackmodellering och (7) Risk- och påverkananalys. Genom dessa steg möjliggör PASTA en djupgående förståelse för hur en angripare kan gå tillväga för att kompromettera ett system, samt vilka konsekvenser det skulle få för verksamheten. Till skillnad från mer statiska modeller, använder PASTA en dynamisk och iterativ process som även innefattar simulering av attacker, vilket gör den särskilt användbar i komplexa miljöer med höga säkerhetskrav (Practical Devsecops 2023).

3. METOD

Detta kapitel beskriver den metodik som använts för att genomföra studien, med fokus på hur penetrationstester har tillämpats för att identifiera säkerhetsbrister i nätverksanslutna enheter. Valet av metod har styrts av studiens syfte, att empiriskt undersöka enheternas motståndskraft mot cyberhot i relation till EU:s aktuella cybersäkerhetsreglering. Kapitlet redogör för den valda forskningsstrategin, hur urvalet av enheter har gjorts, vad penetrationstestning som metod innebär, samt hur datainsamling och analys har genomförts.

3.1 Forskningsmetod

Denna studie använder en forskningsmetod där penetrationstester utförs på utvalda enheter för att identifiera sårbarheter och bedöma efterlevnaden av EU:s cybersäkerhetskrav. Studien kombinerar en kvantitativ analys av identifierade sårbarheter med en kvalitativ bedömning av de potentiella konsekvenserna av dessa sårbarheter.

För att förstå de säkerhetsmässiga aspekterna av enheterna som testas, valdes en blandad metodansats som kombinerar både kvantitativa och kvalitativa metoder. Genom att använda dessa två metoder parallellt ges en mer heltäckande bild av enheternas säkerhet, vilket möjliggör en mer nyanserad förståelse för de tekniska och regulatoriska sårbarheterna. Den kvantitativa delen fokuserar på att samla mätbara och numeriska data, medan den kvalitativa delen möjliggör en djupare tolkning av dessa data i relation till de relevanta EU-direktiven och deras krav på cybersäkerhet.

3.1.1 Kvantitativ metod

I denna forskning valdes ett kvantitativt forskningsupplägg eftersom syftet var att mäta, identifiera och utvärdera konkreta tekniska sårbarheter i olika enheter genom strukturerade penetrationstester. De data som samlades in, såsom öppna portar, svarstider vid belastningstest, samt förekomst av specifika sårbarheter är numerisk och mätbar, vilket är kännetecknande för kvantitativa metoder. Enligt "Seminar on Qualitative Research and Quantitative Research in Social and Human Sciences" kännetecknas kvantitativ forskning av strukturerade datainsamlingsmetoder, fokus på orsak–verkan-relationer och möjligheten att validera samband genom statistiska tekniker (Rashid 2023). Denna typ av metod lämpar sig särskilt väl i tekniska sammanhang där resultaten är numeriska och generaliserbara, vilket är i linje med forskningens syfte.

Genom att utföra detaljerade penetrationstester på varje enhet, kartlades potentiella säkerhetshål som exempelvis onödigt exponerade portar eller specifika sårbarheter som skulle kunna utnyttjas av angripare. Genom att analysera de insamlade kvantitativa data kunde en detaljerad bedömning göras av de sårbarheter som identifierades, vilket direkt relaterade till studiens forskningsfråga om vilka typer av sårbarheter som förekommer i de testade enheterna.

3.1.2 Kvalitativ metod

Trots den kvantitativa fokusen förekommer även inslag av kvalitativa bedömningar, eftersom inte all data är direkt numerisk eller lätt att kvantifiera. I vissa fall, som exempelvis vid observation av öppna portar eller obegränsade inloggningsförsök, framkom inga direkt mätbara värden, men dessa kunde ändå indikera potentiella säkerhetsbrister. För att hantera dessa situationer, där data inte alltid kan mätas numeriskt, krävs en tolkning där man bedömer om dessa beteenden är acceptabla utifrån enhetens funktion eller enligt tidigare forskning.

Kvalitativ forskning kännetecknas av att den fokuserar på att tolka fenomen i sina naturliga sammanhang och hantera frågor som inte alltid är möjliga att kvantifiera matematiskt (Rashid 2023). I detta fall innebär det att resultaten från penetrationstester tolkas i relation till cybersäkerhetskraven i EU:s direktiv, och bedömningen av sårbarheterna görs utifrån deras potentiella påverkan på enheternas överensstämmelse med dessa krav.

Eftersom kvalitativa bedömningar ofta bygger på tolkning kan resultatet påverkas av bedömarens tekniska erfarenhet. För att motverka detta, återkopplades varje tolkning till specifika formuleringar i de relevanta EU-direktiven snarare än att baseras på subjektiva uppfattningar. Detta ökade studiens trovärdighet och minskade risken för snedvridning i analysen.

3.1.3 Metodtriangulering

För att stärka studiens trovärdighet och ge en mer heltäckande bedömning av de testade enheternas säkerhet användes en metodtriangulering, där både kvantitativa och kvalitativa metoder kombinerades. Genom att analysera de tekniska sårbarheterna numeriskt (kvantitativt) och samtidigt tolka dessa resultat i relation till EU:s cybersäkerhetskrav (kvalitativt) skapades en mer robust och nyanserad bild av enheternas säkerhetsstatus. Denna kombination gör det möjligt att fånga både de konkreta, mätbara aspekterna av säkerheten och de mer komplexa, kontextuella aspekterna som handlar om hur dessa sårbarheter påverkar efterlevnaden av lagstiftningen.

Den kvantitativa delen av metoden möjliggjorde en systematisk identifiering av tekniska sårbarheter, medan den kvalitativa bedömningen användes för att tolka resultaten i relation till de regulatoriska kraven och ge en bredare förståelse av sårbarheternas potentiella påverkan. På detta sätt förstärker metoden både precisionen i mätningarna och djupet i tolkningarna, vilket gör det möjligt att dra slutsatser om de tekniska och regulatoriska implikationerna av de identifierade sårbarheterna.

3.1.4 Penetrationstest som metod

Penetrationstest är en systematisk metod för att identifiera, utnyttja och dokumentera sårbarheter i digitala system. Testet simulerar faktiska angreppsscenarier med syfte att utvärdera ett systems motståndskraft mot cyberhot. Enligt NIST är penetrationstester en form av teknisk säkerhetsgranskning där man genom praktiska attacker försöker identifiera sårbarheter som annars inte skulle upptäckas med enbart sårbarhetsskanning eller granskning av dokumentation (NIST u.å).

Penetrationstester utförs ofta i olika faser, rekognosering, skanning, exploatering och rapportering, och bygger på etablerade metodramverk såsom OWASP Testing Guide eller PTES (Penetration Testing Execution Standard). I den här studien fokuseras särskilt på teknisk kartläggning, webbsårbarheter och systemens beteende vid DoS-liknande belastningstester. Denna metodik gör det möjligt att empiriskt bedöma hur väl enheterna lever upp till säkerhetskraven i exempelvis NIS2, CRA och RED DA-direktiven (PTES 2014).

3.2 Förberedelse för labbmiljö

En stor del av experimentet bestod i att genomföra en litteraturstudie kring de verktyg och sårbarheter som är relevanta vid penetrationstestning. Detta inkluderade att undersöka hur olika verktyg fungerar, vilka typer av sårbarheter som är vanliga i specifika typer av enheter, samt vilka metoder som används för att identifiera och hantera dessa sårbarheter. Litteraturstudien var avgörande för att kunna förbereda sig inför laborationerna på ett strukturerat och informerat

sätt. Den gav både en teoretisk grund och praktiska insikter som bidrog till att minska risken för misstag under själva testförfarandet. Genom att på förhand förstå potentiella utmaningar kunde mer tid läggas på själva genomförandet av testerna snarare än felsökning eller omplanering.

Labbmiljön krävde endast en fysisk dator med en virtuell maskin samt den enhet som skulle testas. Vissa enheter krävde en fysisk Ethernet-kabel för att ansluta till den virtuella maskinen, medan andra kunde anslutas trådlöst.

Den virtuella maskinen som användes var Kali Linux, körd genom Oracle VirtualBox. En virtuell miljö valdes eftersom den är enkel att installera, mer portabel än en fysisk lösning och ger en högre säkerhetsnivå. Om något skulle gå fel under penetrationstesterna påverkas enbart den virtuella miljön, inte den fysiska datorn.

Kali Linux valdes specifikt eftersom det är anpassat för penetrationstestning och sårbarhetsanalys. Det innehåller ett brett utbud av verktyg för dessa ändamål, vilket gör testprocessen både smidigare och mer effektiv (Kali Linux 2024).

3.2.1 Installation och versioner

De program som har använts är:

- Kali Rolling (2024.4) x64 – (2024-11-30)
  - 4096MB RAM
  - 4 processor kärnor
- Nmap 7.94SVN
- Gobuster 3.6
- Wfuzz 3.1.0
- Msfconsole 6.4.34-dev
- Wireshark 4.4.0
- Hydra v9.5 (c) 2023
- Netcat v1.10-48.2
- Zaproxy 23.0.1
- Burpsuite 2024.9.4-33549
- Hping3 3.0.0-alpha-2

3.3 Urval av enheter

De enheter som undersöks i denna studie är valda av företaget baserat på deras relevans inom IT- och OT-miljöer samt deras potentiella exponering för cybersäkerhetshot. Urvalet har baserats på följande kriterier:

- En hög förekomst i industriella och kommersiella sammanhang
- Funktionalitet för nätverksanslutning och fjärråtkomst
- Relevans i förhållande till krav och bestämmelser enligt NIS2-direktivet, RED DA samt CRA

Dessa faktorer har legat till grund för att identifiera enheter med särskilt hög riskprofil och som därmed är av särskilt intresse vid en säkerhetsgranskning i linje med EU:s cybersäk erhetsreglering. Eftersom flera av enheterna ännu inte lanserats på marknaden och är föremål för sekretess, har enheterna anonymiserats i denna rapport. De refereras istället till som Enhet A–G i de följande avsnitten.

3.3.1 Enhet A

Enhet A är en fysisk enhet som är ansluten via Ethernet och tillhandahåller ett webbaserat gränssnitt för kommunikation och kontroll. Tjänsten är tillgänglig via HTTP på port 80 och är endast åtkomlig genom den lokala nätverksinfrastrukturen. Systemet körs på ett RTOS (Real Time Operating System) som är optimerat för inbyggda tillämpningar, som är specialiserade program som körs på enheter som inte är vanliga datorer, till exempel i en bil, en tvättmaskin eller en nätverksrouter. De är ofta optimerade för att göra en viss uppgift effektivt och pålitligt.

Testomfånget för enheten inkluderar operativsystemidentifiering, fullständig portskanning för att säkerställa att endast de avsedda tjänsterna är öppna, samt genomförande av DoS-tester för att utvärdera enhetens sårbarhet mot överbelastningsattacker. Penetrationstestverktyg ska användas för att analysera de öppna portarna och säkerställa att inga okända säkerhetsrisker finns. Kontroll mot kända sårbarheter (CVE:er) är en del av testet för att identifiera eventuella tidigare kända svagheter.

En viktig kommentar är att ingen annan port än den specificerade får vara öppen, vilket innebär att portskanningar måste genomföras med noggrannhet för att säkerställa att enheten endast exponerar nödvändiga tjänster.

3.3.2 Enhet B

Enhet B är en specialiserad inbyggd plattform som använder Ethernet-anslutning för åtkomst och hantering. Det kör ett Linux-baserat operativsystem och är tillgängligt via SSH på port 22. Kommunikationen sker över TCP/IP, och systemet är designat för att vara flexibelt, med stöd för ett brett utbud av hårdvaruarkitekturer, vilket gör det lämpligt för en mängd inbyggda tillämpningar.

Testomfånget inkluderar operativsystemidentifiering, en komplett portskanning för att identifiera alla öppna tjänster samt genomförande av DoS-simuleringar för att utvärdera systemets motståndskraft mot potentiella överbelastningar. Penetrationstestverktyg ska användas på alla öppna portar för att säkerställa att eventuella sårbarheter identifieras. Dessutom ska kända CVE:er kontrolleras för att upptäcka eventuella tidigare dokumenterade säkerhetsbrister.

En viktig kommentar är att alla öppna portar ska listas, och att inga oönskade tjänster får vara aktiva. Portskanningar ska specifikt rikta sig mot de kända öppna portarna och verifiera att inga oväntade portar exponeras.

3.3.3 Enhet C

Enhet C är tillgängligt via det publika nätet och möts initialt av en inloggningssida där åtkomst kräver användarnamn och lösenord. Denna åtkomstkontroll utgör det första säkerhetslagret mot obehörig åtkomst till funktioner och data. Kommunikation sker via TCP/IP och systemets webbaserade gränssnitt är åtkomligt på port 443 via en DNS-adress. Det är viktigt att notera att tjänsten ligger bakom en webbproxy, vilket innebär att endast DNS-namnet får användas vid testning och direkt åtkomst via IP-adress är inte tillåten.

Systemet körs på ett Linux-baserat operativsystem och testomfånget omfattar identifiering av operativsystem, samt användning av penetrationstestverktyg riktade enbart mot den angivna HTTPS-adressen. Det inkluderar även kontroller mot kända sårbarheter (CVE:er). Avgränsningar i testet innebär att endast specificerade tjänster får analyseras, där bland annat fullständig portskanning eller användning av IP-adress uttryckligen inte är tillåten.

3.3.4 Enhet D

Enhet D utgör en lokalt installerad instans av ett versionshanteringssystem, vilket tillhandahåller funktionalitet för kodhantering, kontinuerlig integration och distribution (CI/CD) samt samarbetsstöd mellan utvecklare. Systemet är isolerat till det interna nätverket och är därmed inte exponerat mot internet. Åtkomst sker via HTTP på port 80 genom direkt IP-adress, och all kommunikation sker över TCP/IP-protokollet.

Systemet är konfigurerat för att köra i en Linux-baserad miljö och omfattas av följande testmoment: identifiering av operativsystem, fullständig portskanning, sårbarhetsanalys baserat på kända CVE:er, samt stresstester i form av DoS-simuleringar. Samtliga öppna portar ska identifieras och dokumenteras. Testningen begränsas till den specifika IP-adress där instansen är åtkomlig, vilket möjliggör portscanning och trafikinspektion utan krav på DNS-upplösning eller omvänd proxy.

Som en del av angreppsytans utvärdering har testgruppen erhållit autentiseringsuppgifter i form av användarnamn och lösenord. Detta möjliggör en riktad brute-force-attack i syfte att bedöma systemets motståndskraft mot obehörig åtkomst genom upprepade inloggningsförsök.

3.3.5 Enhet E

Systemet är en lokal installation av en filsynkroniserings- och delningstjänst som möjliggör lagring, åtkomst och samarbete kring filer i en lokal miljö. Tjänsten är inte tillgänglig via det publika nätet, utan nås enbart inom det lokala nätverket via HTTP på port 8080, genom direkt IP-åtkomst. Kommunikation sker över TCP/IP och systemet är baserat på ett Linux-operativsystem.

Testomfånget inkluderar identifiering av operativsystem, fullständig portskanning, kontroll av kända sårbarheter (CVE:er), samt genomförande av DoS-tester i syfte att utvärdera systemets motståndskraft mot överbelastningsattacker. Alla öppna portar ska identifieras och dokumenteras, inklusive resultat från portskanningar.

3.3.6 Enhet F

Enhet F är en nätverksansluten dörrtelefon som använder Ethernet för kommunikation och fjärråtkomst. Den kör ett Linux-baserat operativsystem och är tillgänglig via SSH på port 22. Enheten kan nås genom den specificerade IP-adressen, och all kommunikation sker över TCP/IP.

Testomfånget inkluderar operativsystemidentifiering, en fullständig portskanning för att identifiera alla öppna tjänster samt genomförande av DoS-tester för att utvärdera enhetens sårbarhet för överbelastningsattacker. Penetrationstestverktyg ska användas för att analysera de öppna portarna och kontrollera för eventuella säkerhetsbrister. Dessutom ska kända CVE:er kontrolleras för att säkerställa att inga tidigare dokumenterade sårbarheter existerar.

En viktig kommentar är att alla öppna portar ska listas och att endast de specificerade portarna får vara öppna. Portskanningarna ska riktas mot de öppna portar som har identifierats och säkerställa att inga oväntade portar exponeras.

3.3.7 Enhet G

Enhet G är en central nätverksansluten enhet som fungerar som knutpunkt för kommunikation mellan flera andra enheter i nätverket, inklusive dörrsystem och andra proprietära komponenter. Enheten är ansluten via Ethernet och erbjuder fjärråtkomst via SSH på port 22. Kommunikation

sker över TCP/IP och operativsystemet är Linux-baserat, vilket möjliggör flexibel hantering och integration.

Testomfånget innefattar identifiering av operativsystem, fullständig portskanning för att kartlägga exponerade tjänster, samt DoS-tester för att utvärdera systemets motståndskraft vid överbelastning. Penetrationstestverktyg ska användas på alla öppna portar för att identifiera eventuella säkerhetsbrister. Kontroll mot kända CVE:er ingår i analysen för att upptäcka sårbarheter som redan dokumenterats i offentliga databaser.

Det är särskilt viktigt att samtliga öppna portar dokumenteras noggrant, och att skanningar bekräftar att inga obehöriga eller oväntade tjänster är tillgängliga från nätverket.

3.4 Genomförande av penetrationstest

Genomförandet av penetrationstestet har delats in i tre huvudfaser: planering/recon, skanning och sårbarhetsanalys. Denna struktur speglar etablerade metoder inom cybersäkerhet och penetrationstestning, såsom de som föreskrivs av PTES. Samtliga moment har som syfte att kartlägga, analysera och pröva enheternas motståndskraft mot attacker, i linje med cybersäkerhetskraven i NIS2 och CRA (PTES 2014).

3.4.1 Planering/Recon

I det inledande steget samlas information om de enheter som ska testas, med särskilt fokus på kända sårbarheter. Detta görs genom att söka i CVE-databasen efter relevanta identifierade svagheter. Målet är att skapa en uppdaterad hotbildsanalys för varje testobjekt, vilket är i linje med kraven på riskhantering i NIS2-direktivet. Denna fas är avgörande för att säkerställa att testningen fokuserar på verkliga och relevanta hot.

Denna planeringsfas är också viktig eftersom den möjliggör en mer effektiv resursfördelning under testningen. Genom att identifiera de mest sannolika och kritiska sårbarheterna i förväg kan testarna prioritera insatserna där de gör störst nytta. Det minskar risken för att tid läggs på irrelevanta eller lågprioriterade områden och ökar samtidigt chansen att upptäcka säkerhetsbrister som faktiskt kan utnyttjas i praktiken. Detta tillvägagångssätt är särskilt värdefullt i komplexa miljöer med många olika enheter, där det inte är realistiskt att genomföra fullständiga tester på varje aspekt.

3.4.2 Skanning

Själva penetrationstestet inleds med en sårbarhetsskanning, där verktyg som Nmap används för att identifiera öppna portar och avgöra om någon av dessa inte borde vara tillgänglig. Därefter analyseras enhetens webbgränssnitt genom att granska dess källkod. Syftet är att identifiera potentiellt känslig information som oavsiktligt har gjorts tillgänglig, såsom API-nycklar, hårdkodade inloggningsuppgifter eller dolda kataloger (Nmap u.å).

En annan viktig del av testet är att försöka extrahera favicon-ikonen från webbplatsens källkod. Detta kan göras manuellt eller automatiseras med verktyget GoBuster eller WFuzz, som används för att identifiera underkataloger på webbplatsen. Om favicon-filen hittas kan dess MD5-hashvärde genereras och jämföras med den offentliga databasen OWASP_favicon_database. Även om detta inte i sig är en sårbarhet, kan det avslöja vilken typ av applikationer eller teknologier som används på enheten, vilket ger värdefull information inför vidare testning. Detta moment handlar därför mer om informationsinhämtning än om att identifiera direkta sårbarheter (Johari, KARTHIKRAM, Kumar, Reddy & Kumar, 2024).

3.4.3 Sårbarhetsanalys

För att bedöma enhetens motståndskraft mot en DoS-attack genomförs en belastningstestning med hjälp av hping3. Attacker genomförs med ICMP-paket och SYN-flooding för att se hur enheten hanterar stora mängder trafik. Testet utvärderar om webbgränssnittet blir otillgängligt, prestandan försämras eller om systemet har några skyddsmekanismer på plats för att hantera sådana attacker.

Slutligen används ZAP Proxy, en open-source web application scanner, för att identifiera vanliga webbsårbarheter enligt OWASP Top 10, såsom XSS (Cross-Site Scripting), SQL Injection, CSRF (Cross-Site Request Forgery) och insecure deserialization. ZAP simulerar attacker mot webbplattformens olika inmatningspunkter och analyserar responsen för att identifiera potentiella svagheter som kan utnyttjas av en angripare.

3.5 Riskbedömning

För att bedöma riskerna kopplade till de sårbarheter som identifierades under penetrationstesterna användes hotmodellen DREAD. Denna modell gör det möjligt att systematiskt analysera varje sårbarhet utifrån fem centrala faktorer, vilket underlättar en jämförelse och prioritering av risknivåer. Genom att tilldela poäng till varje kategori, till exempel hur stor skada sårbarheten kan orsaka, hur lätt den är att utnyttja, och hur många användare som påverkas, kan man få en tydligare helhetsbild av hotets allvarlighetsgrad. Detta tillvägagångssätt gör det enklare att identifiera vilka sårbarheter som bör åtgärdas först och vilka som eventuellt kan hanteras med förebyggande åtgärder på längre sikt.

För att säkerställa en enhetlig och motiverad poängsättning användes följande riktlinjer enligt (Yuga 2022):

- Damage Potential: Hur allvarliga konsekvenser ett utnyttjat hot kan få.
    - 0: Ingen skada
    - 5: Informationsläckage
    - 8: Icke-känsliga persondata exponeras
    - 9: Administrativ data exponeras
    - 10: Total systemnedgång, datadestruktion eller otillgänglighet
- Reproducibility: Hur lätt en attack kan upprepas.
    - 0: Nästan omöjlig eller mycket svår
    - 5: Kräver komplexa steg
    - 7.5: Relativt enkel
    - 10: Mycket enkel
- Exploitability: Hur lätt det är att praktiskt genomföra attacken.
    - 2.5: Kräver avancerade tekniska färdigheter
    - 5: Kräver tillgängliga verktyg
    - 9: Kräver applikationsproxys
    - 10: Kräver enbart en webbläsare
- Affected Users: Hur många användare som påverkas.
    - 0: Inga användare
    - 2.5: Enskilda användare
    - 6: Flera användare
    - 8: Administrativa användare
    - 10: Alla användare
- Discoverability: Hur lätt det är att upptäcka sårbarheten.
    - 0: Svårupptäckt
    - 5: Kan hittas via öppna förfrågningar

- 8: Offentligt känd eller sökbar
- 10: Mycket lätt att hitta, exempelvis via öppen form eller synlig URL

Detta tillvägagångssätt ger en kvantitativ grund för att jämföra och prioritera åtgärder utifrån vilka hot som är mest kritiska. Riskbedömningen presenteras och analyseras per enhet i avsnitt 5.3.

4. RESULTAT

Detta kapitel presenterar resultaten från de praktiska penetrationstester som genomförts mot sju olika nätverksanslutna enheter. Då enheterna är säkerhetsklassificerade anonymiseras deras identitet genom beteckningarna Enhet A till Enhet G. Även IP-adresser ersätts med termen ”IP-adress” och under testen gång förekommer även företagsnamn som ersätts med ”Företag AB”. Testerna fokuserade på att identifiera sårbarheter relaterade till exponerade tjänster, autentiseringsmekanismer och motståndskraft mot överbelastningsattacker, i syfte att bedöma enheternas säkerhetsnivå i förhållande till gällande cybersäkerhetskrav.

4.1 Empiri På Utfört Experiment

4.1.1 Enhet A

Enhet A är en inbyggd nätverksansluten enhet med ett webbaserat gränssnitt som används för lokal administration. Under testförhållandena var den direkt ansluten till en testdator via Ethernet, vilket möjliggjorde isolerad åtkomst utan koppling till externa nätverk. Gränssnittet exponerades via HTTP och fungerade som primär åtkomstpunkt för konfigurationsändamål. Denna uppsättning gav en kontrollerad miljö för att undersöka enhetens beteende vid olika säkerhetstester.

Företaget som tillhandahåller enheten har specificerat att den är baserad på operativsystemet FreeRTOS, som ett första steg inleddes därför testningen med att granska publika CVE-listor för FreeRTOS i syfte att identifiera eventuella kända sårbarheter. Denna granskning resulterade dock inte i några relevanta fynd som kunde användas i det fortsatta arbetet.

Sedan inleddes en teknisk kartläggning av enheten genom nätverksskanning med verktyget Nmap. Med hjälp av olika parametrar (bland annat -sV för tjänstidentifiering, -A för avancerad detektion, -O för operativsystemsanalys och -p- för att skanna alla portar) identifierades vilka nätverksportar som var öppna och vilka tjänster som kördes (Nmap u.å). Resultatet visade att endast port 80 (HTTP) var öppen, vilket indikerar att enheten har ett webbgränssnitt. Det gick däremot inte att med säkerhet fastställa vilket operativsystem som enheten använder.

För att undersöka webbtjänstens innehåll användes webbläsarens utvecklarverktyg, där både källkod och nätverkstrafik granskades. Genom detta arbete identifierades flera tillgängliga resurser, såsom /favicon.ico, /global.css, /build/bundle.css, /manifest.json, /icon512.png och /polyfill.js. Dessa filer analyserades för att få en bättre bild av tjänstens struktur och möjliga attackytor. Resursen /favicon.ico hämtades separat och dess hashvärde genererades med hjälp av curl och md5sum. Det genererade värdet jämfördes mot OWASP:s favicon-databas i syfte att eventuellt identifiera underliggande ramverk eller teknologier, men ingen matchning kunde påvisas. Analysen av övriga resurser resulterade inte i några identifierade relevanta sårbarheter.

Vidare genomfördes en bruteforce-attack med verktyget Gobuster i kombination med en välkänd ordlista från Dirb-projektet, för att identifiera dolda kataloger och filer. För att upptäcka potentiella sårbarheter utfördes även en automatiserad skanning med Nikto, men försöken gav inga signifikanta resultat.

För att utvärdera enhetens motståndskraft mot DoS-attacker genomfördes flera tester. Dessa inkluderade ICMP-baserade attacker via hping3, vilket resulterade i att tjänsten blev otillgängligt men hemsidan kunde laddas in. Ytterligare attacker i form av SYN-floods genomfördes, vilket medförde att tjänsten frös och inte längre svarade på anrop.

Slutligen användes säkerhetsverktygen OWASP ZAP och Burp Suite för att genomföra såväl automatiserade som manuella skanningar. Burp Suite användes även för att utföra injektionstester och analysera WebSocket-kommunikation för att identifiera eventuella logiska eller tekniska brister i hanteringen av klient-serverinteraktioner, dessa tester visade inga sårbarheter.

4.1.2 Enhet B

Enhet B är en kraftfull inbyggd enhet som använder Ethernet för nätverkskommunikation. Den är byggd för tekniskt avancerade tillämpningar och erbjuder åtkomst via ett textbaserat gränssnitt genom SSH. Till skillnad från enklare enheter bygger den på ett fullfjädrat Linux-operativsystem, vilket ger större flexibilitet men också en potentiellt bredare angreppsyta. Under testningen utvärderades dess säkerhetsegenskaper i en isolerad miljö med fokus på tjänsteexponering och motståndskraft mot nätverksbaserade attacker.

Testproceduren inleddes med en omfattande nätverksskanning med Nmap, vilken identifierade att portarna 22, 111, 1534 och 5355 var öppna. Dessutom genomfördes en operativsystemdetektering som indikerade att enheten kör en Linux-kärna inom versionerna 4.15 till 5.8. För att utvärdera de potentiella sårbarheterna i den aktuella Linuxversionen konsulterades relevanta CVE-databaser med särskilt fokus på Linux 5.8. Sökningen resulterade i tio identifierade sårbarheter, men ingen av dessa bedömdes vara tillämplig på den testade enheten.

Vidare utfördes en Nmap-baserad sårbarhetsskanning för att identifiera eventuella konfigurationsfel eller kända säkerhetsbrister i systemet. Port 1534 (micromuse-lm) inte är en standardport med väldefinierad tjänst och oftast används av proprietära eller interna applikationer, port 5355 är kopplad till LLMNR, en protokolltyp för lokal namnupplösning.

Som en del av säkerhetsanalysen undersöktes även port 22 (SSH) med hjälp av Netcat för att verifiera att tjänsten var aktiv och för att identifiera vilken SSH-server. Vilket avslöjade sig vara SSH-2.0-dropbear_2022.83 (en lättvikt SSH-server), samt vilka kryptografiska algoritmer som stöddes. Insamlad information analyserades i syfte att identifiera svagheter, men analysen resulterade inte i några relevanta fynd.

En inledande undersökning riktades mot port 111, vilket är associerad med RPC (Remote Procedure Call)-tjänster. För att kartlägga vilka RPC-tjänster som var aktiva användes kommandot *rpcinfo -p*, vilket gav en överblick över tillgängliga program och portnummer. Därefter för att undersöka systemets robusthet mot överbelastningsattacker genomfördes ett flertal tester med verktyget msfconsole, där modulen *RPCBOMB* användes. Den aktuella sårbarheten identifierades initialt genom en sökning i en CVE-databas (CVE-2017-8779) och verifierades genom praktisk testning. Under testets gång observerades en temporär försämring av systemets responstid, vilket tyder på en viss sårbarhet mot överbelastning i denna del av systemet. Resultaten visar att systemet uppvisar påverkbarhet vid specifika överbelastningsförsök mot RPC-portar, särskilt via port 111, vilket motiverar vidare analys av tjänstens säkerhetskonfiguration och möjligheten till härdning mot liknande attacker.

För att utvärdera systemets motståndskraft mot autentiseringsattacker skapades en ordlista med potentiella användarnamn, baserat på vanliga namn och mönster. En brute-force-attack genomfördes därefter mot SSH-tjänsten med hjälp av verktyget Hydra. Ordlistan för lösenorden utgjordes av ett utdrag från komprometterade lösenord från Shein-läckan (Gustafsson 2022). Syftet med denna fas var att analysera om systemet använde standardiserade eller svaga

autentiseringsuppgifter. Brute-force-attacken pågick i 30 minuter men resulterade inte i någon lyckad inloggning.

Ytterligare tester genomfördes med hping3, där både en SYN flood och en ICMP flood-attack implementerades. Vid båda attackerna observerades en utdragen minskning i prestanda med långsamma responstider, vilket indikerar att systemet påverkas negativt av hög trafikbelastning under attackerande förhållanden.

4.1.3 Enhet C

Enhet C är exponerad mot det publika nätet och tillhandahåller ett webbaserat gränssnitt som skyddas av autentisering. Den nås enbart via en DNS-adress och använder krypterad kommunikation (HTTPS). Enheten är placerad bakom en webbproxy, vilket begränsar möjligheterna till direkt teknisk åtkomst och kräver testning via angiven URL. Dess externa exponering gör den särskilt intressant ur ett cybersäkerhetsperspektiv.

Testet inleddes med att undersöka autentiseringsfunktionens robusthet genom att försöka logga in med generiska och vanligt förekommande användarnamn och lösenord. Ingen åtkomst erhölls, men syftet var att avgöra om systemet svarade med spärr eller andra skyddsåtgärder vid upprepade inloggningsförsök. Inga sådana skydd kunde observeras, vilket tyder på att systemet saknar mekanismer för att begränsa brute-force-attacker, en potentiell säkerhetsbrist.

Därefter utfördes en Nmap-skanning riktad mot port 443(https) för att samla information om den bakomliggande tjänsten samt det underliggande operativsystemet. Skanningen visade att webbservern använde *Apache httpd 2.4.62*, men kunde inte identifiera något specifikt operativsystem. CVE-skanning gjordes på *Apache httpd 2.4.62,* dock hittades inga kända sårbarheter.

Som ett led i att identifiera eventuell bakomliggande teknik extraherades webbapplikationens favicon och dess MD5-hash genererades. Denna jämfördes med OWASP:s favicon-databas, dock utan träff. Detta tyder på att applikationen inte omedelbart går att koppla till något allmänt känt ramverk eller system baserat på favicon-hashning.

Avslutningsvis användes verktyget Wfuzz för att genomföra en katalogskanning av webbapplikationen. Denna skanning resulterade i att webbservern kraschade och förblev otillgänglig vid flera försök att återansluta, både direkt och via olika IP-adresser genom VPN. Detta tyder på att servern har begränsad kapacitet att hantera intensiva förfrågningar eller att otillräcklig felhantering förekommer, vilket potentiellt kan utnyttjas för att orsaka en tillgänglighetsstörning.

Eftersom webbservern förblev otillgänglig och krävde en omstart i kombination med en uppdatering för att återställas till funktionellt tillstånd, beslutades det att avbryta vidare felsökning och istället fortsätta med efterföljande moment i laborationen.

4.1.4 Enhet D

Denna enhet representerar ett internt versionshanteringssystem som används för mjukvaruutveckling och samarbeten. Den är endast åtkomlig inom det lokala nätverket och är uppsatt på en IP-adress utan DNS-stöd. Enhet D erbjuder funktionalitet för kodlagring, CI/CD-pipelines och autentiserad användaråtkomst. Dess roll i utvecklingskedjan gör den kritisk ur både säkerhets- och driftsynpunkt.

Först genomfördes en nätverksskanning med Nmap mot enheten för att kartlägga öppna portar och identifiera aktiva tjänster. Resultatet av skanningen visade att portarna 22 (SSH), 25 (SMTP), 80 (HTTP), 8060 (HTTP), 9094 (okänd tjänst) och 33441 (HTTP – Go-IPFS JSON-RPC eller InfluxDB API) var öppna.

SSH-tjänsten på port 22 kör OpenSSH 9.2p1 på Debian, vilket tyder på att en relativt ny version används. Port 25 var associerad med Postfix SMTP, en vanlig mailserver. Port 80 körde en HTTP-server utan angiven version, medan port 8060 körde nginx 1.27.4 en modern och säker version av den populära webbservern. Port 33441 svarade med Golang net/http server, vilket kan indikera att en applikation byggd i Go kör ett API- eller backend-gränssnitt.

Port 9094 svarade, men tjänsten kunde inte identifieras av Nmap. Detta kan tyda på antingen en proprietärtjänst eller en som inte annonserar sig enligt standard.

Operativsystemdetektering indikerade att enheten kör en Linux-distribution, troligen baserad på kärnversion 4.x eller 5.x. Baserat på övriga indikatorer kunde enheten klassificeras som en generell systemtyp eller potentiellt ett inbyggt system, t.ex. en NAS-enhet eller brandvägg. Vid efterföljande CVE-skanningar av de öppna tjänsterna och deras respektive protokoll noterades att port 80 och 8060 inte uppvisade några kända sårbarheter, då tjänsterna kördes med de senaste uppdateringarna.

Vidare tillämpades säkerhetsverktygen OWASP ZAP och Nikto för att automatiskt analysera webbapplikationen. Med OWASP ZAP genomfördes en aktiv skanning för att identifiera potentiella sårbarheter i applikationens struktur och beteende, medan Nikto användes för att upptäcka osäkra filer, gamla versioner av mjukvara samt potentiellt farliga konfigurationsinställningar. Kombinationen av dessa verktyg möjliggjorde en bredare kartläggning av applikationens exponerade ytor dock upptäcktes inga sårbarheter.

Ytterligare en omfattande undersökning utfördes med Wfuzz, vilket ledde till upptäckten av ett administratörskonto med användarnamnet "@root" samt en tillgänglig robots.txt-fil som kunde ge ytterligare information om applikationens struktur. Tester genomfördes även genom att använda Netcat för att analysera en banner grabbing där svaret indikerade *"220 smtp.IP-adress ESMTP Postfix (Debian/GNU)"*. Svaret användes för att genomföra en riktad kontroll i CVE-databasen för att identifiera eventuella kända sårbarheter kopplade till den identifierade versionen av Postfix, vilket gav två träffar med utdaterad information (CVE-2015-2775 och CVE-2009-2939).

Därefter utfördes ett antal funktionstester för att verifiera att autentiseringsmekanismerna inte omfattade någon automatisk blockering vid upprepade inloggningsförsök. Detta uppnåddes genom att upprepade gånger logga in i systemets webbgränssnitt, vilket visade att ingen blockering aktiverades trots flera försök. För att undersöka om autentiseringssystemet var sårbart för brute-force-attacker användes verktyget Hydra i syfte att försöka erhålla inloggningsuppgifterna för administratörskontot ”root”. Därefter genomfördes ett kompletterande test mot användaren ”marc”, där lösenordet var känt i förväg. Det kända lösenordet inkluderades i en ordlista bestående av totalt 15 olika lösenord för att utvärdera om verktyget kunde identifiera det korrekta alternativet. Trots dessa försök kunde inget entydigt resultat erhållas, och lösenordet utmärktes inte tydligt under attackförsöket. Det aktuella angreppet bedöms därför som ineffektivt under givna förutsättningar.

Avslutningsvis gjordes DoS-tester där hping3 utnyttjades för att sända ICMP- och SYN-paket. Under ICMP-testet fortsatte webbapplikationen att fungera normalt, medan en SYN-flood attack ledde till en påtaglig nedgång i prestanda, vilket resulterade i en långsam responstid.

4.1.5 Enhet E

Enhet E är en självhostad lösning för fildelning och synkronisering i en lokal miljö. Den erbjuder ett webbaserat gränssnitt på en icke-standardport och är isolerad från externa nätverk. Systemet är designat för intern samverkan och tillgång till dokument, vilket gör det till en central komponent för datalagring inom organisationens nätverk.

Den inledande fasen omfattade en portscanning med Nmap mot enheten, där samtliga portar granskades. Denna skanning visade att portarna 22(SSH), 1883(mosquitto) och 8080(Apache HTTPD) var öppna, vilket gav en första indikation på vilka tjänster som exponerades. Därefter genomfördes en granskning av kända CVE-rapporter med särskilt fokus på sårbarheter associerade med ”mosquitto version 2.0.11” som var tjänsten för port 1883/tcp, för att kunna kartlägga potentiella risker kopplade till tjänsten.

Vidare säkerhetsanalys genomfördes med särskilt fokus på Mosquitto MQTT-brokern (version 2.0.11) på port 1883. En sökning i CVE-databasen visade att denna version är associerad med flera sårbarheter, däribland:

- CVE-2021-41039: som möjliggör DoS (Denial of Service) genom en felaktig hantering av MQTT-paket.

För att utvärdera enhetens motståndskraft mot överbelastningsattacker implementerades flera DoS-tester. En Python-baserad DoS-attack riktades mot MQTT-tjänsten som kördes på enheten, det märktes en försämring i responstiden.

Webbservern på port 8080 rapporterade sig som Apache HTTPD 2.4.62 på Debian, vilket i skrivande stund är en relativt ny version och inte är direkt associerad med kända allvarliga CVE:er.

Utvärderade webbapplikationens inloggningsmekanismer genom att genomföra upprepade inloggningsförsök. Testet visade att systemet har en implementerad begränsning av antalet inloggningsförsök per IP-adress, vilket effektivt förhindrade brute-force-attacker. Även prövades inloggning med de standardanvändarnamn och lösenord som enheten kan vara förkonfigurerad med, utan att någon framgångsrik autentisering uppnåddes.

För att fördjupa säkerhetsanalysen av enheten genomfördes sårbarhetsskanningar med flera kompletterande verktyg. Wfuzz användes för att identifiera dolda kataloger och filer som kan utgöra potentiella angreppsytor. Därefter genomfördes en automatiserad skanning med OWASP ZAP, vilket möjliggjorde en mer omfattande utvärdering av webbgränssnittets sårbarheter.

Vidare hämtades filen för webbapplikationens favicon och dess MD5-hash genererades. Den erhållna hashsumman jämfördes med befintliga databaser för kända hashsummor, utan att någon matchning kunde påvisas.

Avslutningsvisgenomfördes ett ICMP flood-attacker mot webbapplikationen, vilket resulterade i märkbar prestandadegradering trots att tjänsten fortsatte att vara funktionell. Ett påföljande

SYN flood-test riktades mot webbgränssnittet, vilket ledde till att sidan inte längre kunde uppdateras.

4.1.6 Enhet F

Denna enhet är en nätverksansluten dörrtelefon med stöd för fjärradministration via SSH. Den är direkt kopplad till nätverket via Ethernet och erbjuder möjlighet till fjärrkonfiguration och systemintegration. På grund av dess roll i fysisk säkerhet är det viktigt att bedöma både dess nätverksexponering och autentiseringsskydd.

Analysen inleddes med en omfattande portskanning med Nmap, vilket identifierade att endast port 22 (SSH) och port 2033 (glogger) var öppna. För att utvärdera potentiella sårbarheter genomfördes en granskning mot relevanta CVE-databaser. Ingen offentlig sårbarhet kunde identifieras för port 2033, men för port 22 som kör OpenSSH version 8.9 (protocol 2.0) påträffades en känd sårbarhet (CVE-2023-28531).

För att utvärdera autentiseringens robusthet genomfördes en brute-force-attack mot SSH-tjänsten som lyssnade på port 22/tcp. Verktyget Hydra användes tillsammans med en ordlista innehållande vanliga användarnamn och lösenord i syfte att identifiera eventuella exponerade konton. Under attackens gång kunde inga giltiga användarnamn eller lösenord identifieras.

För att utvärdera enhetens hantering av överbelastningsattacker genomfördes flera DoS-tester. Dessa inkluderade:

- En SYN flood-attack, som analyserade enhetens förmåga att hantera ett stort antal samtidiga anslutningsförfrågningar på applikationsnivå. Endast marginella prestandaskillnader kunde observeras.
- En UDP flood-attack, där tjänstens motståndskraft mot ett överflöd av UDP-paket utvärderades. Ingen märkbar påverkan noterades.
- En ICMP flood-attack, vilken resulterade i att enheten fortsatte svara på ICMP-förfrågningar (ping), men uppvisade en tydlig försämring i svarstid under belastning.

4.1.7 Enhet G

Enhet G fungerar som en central hubb i ett större system och ansvarar för kommunikation mellan flera IoT-komponenter, såsom dörrsystem och andra proprietära enheter. Den är nätverksansluten och erbjuder administrativ åtkomst via SSH. Enhetens centrala funktion gör den till en potentiell attackyta med hög påverkan vid en kompromettering.

Innan de mer avancerade testerna påbörjades, genomfördes en portskanning med hjälp av Nmap för att kartlägga enhetens exponerade nätverkstjänster. Resultatet visade att portarna 22 (SSH) och 2034(scoremgr) var öppna, CVE-skanning på scoremgr visade inga kända sårbarheter. Vidare utnyttjades en Nmap-baserad vulnerability scan, vilket identifierade träffar med hänvisning till Företag AB.

För att utvärdera systemets motståndskraft mot överbelastningsattacker genomfördes ett flertal DoS-tester. Dessa tester inkluderade:

- En SYN flood-attack, vilket inte påverkade systemets förmåga att svara på ping-förfrågningar.
- En ICMP flood-attack, som visade liknande resultat, där ping-svaren bibehölls under attacken.
- En UDP flood-attack, där en fortsatt respons på ping-förfrågningar noterades trots den ökade trafikbelastningen.

5. ANALYS

Denna analys fokuserar på att bedöma hur väl de testade enheterna uppfyller säkerhetskraven enligt EU-direktiven NIS2, RED-DA och CRA, baserat på resultaten från genomförda penetrationstester. De viktigaste iakttagelserna från testerna rörde två återkommande sårbarhetskategorier:

1. Exponering för brute-force-attacker utan skydd.
2. Otillräcklig motståndskraft mot DoS-attacker.

Dessa brister är särskilt relevanta i ljuset av direktivens krav:

- **CRA** kräver att digitala produkter har ett inbyggt skydd mot vanliga attacker under hela produktens livscykel. Att flera enheter saknade skydd mot brute-force och föll ihop under relativt enkla DoS-attacker tyder på bristande inbyggd motståndskraft.
- **RED-DA** ställer krav på grundläggande cybersäkerhet hos trådlösa enheter, vilket inkluderar skydd mot obehörig åtkomst. Vår analys visar att bristande skydd mot lösenordsattacker innebär en ökad risk för att obehöriga kan få åtkomst till systemen.
- **NIS2** gäller främst verksamheter av samhällskritisk betydelse. Enheternas sårbarhet för DoS-attacker påverkar tillgänglighet – ett av tre grundkrav i informationssäkerhet och kan därmed innebära att utrustningen inte är lämplig för användning i kritiska system utan ytterligare skydd.

Genom att jämföra varje enhets identifierade sårbarheter med direktivens säkerhetskrav har vi kunnat dra slutsatsen att även om enheterna i viss mån är konfigurerade med säkerhet i åtanke, så finns det tydliga brister som skulle kunna utnyttjas i praktiken. Dessa måste åtgärdas för att uppfylla de krav som ställs av de aktuella EU-direktiven.

5.1 Utmaningar under penetrationstesterna

Att genomföra penetrationstester kan vara utmanande, särskilt utan tidigare erfarenhet. Brist av förkunskaper kan leda till att en stor del av tiden går åt till informationssökning snarare än själva testningen. I detta fall blev därför en betydande del av laborationen att samla in information i förväg, i syfte att förbereda testerna på bästa möjliga sätt. Denna informationssökning omfattade bland annat vilka verktyg som lämpar sig för olika typer av tester, samt vilka typer av sårbarheter eller indikatorer som är särskilt relevanta att identifiera under testförfarandet.

Denna del av arbetet kan kategoriseras som ”Planning and Reconnaissance”, det första steget i en typisk penetrationstestcykel. Under denna fas samlas information om målen för att kunna genomföra tester på ett så effektivt och strukturerat sätt som möjligt. Detta kan innefatta allt från OSINT (Open Source Intelligence), såsom att använda Google för att hitta teknisk dokumentation eller tidigare rapporterade sårbarheter, till att granska CVE-databaser, forum och tillverkares supportsidor. Genom att förstå vilka protokoll, portar och tjänster som används i de aktuella enheterna kan man i förväg avgöra vilka testmetoder som är relevanta. Denna förberedelse är avgörande för att kunna genomföra meningsfulla och målinriktade attacker, och speglar vikten av att kombinera teknisk kompetens med analytisk förmåga redan innan själva testmomentet inleds. En väl genomförd reconnaissance-fas förbättrar inte bara effektiviteten i testningen, utan ökar även chansen att upptäcka mer sofistikerade eller mindre uppenbara sårbarheter (Aibekova & Selvarajah 2022).

En ytterligare utmaning i genomförandet av penetrationstesterna var den begränsade förberedelsetiden. Att testa ett stort antal enheter inom en relativt kort tidsram visade sig vara

krävande, vilket påverkade möjligheten att genomföra djupgående och specifika analyser. Den begränsade tiden ledde till att testerna blev mer övergripande än detaljerade, vilket innebar att fokus främst låg på att identifiera uppenbara och grundläggande säkerhetsbrister snarare än att undersöka enskilda sårbarheter på djupet. Med mer förberedelse och tid hade det varit möjligt att exempelvis analysera specifika sårbarheter som upptäcktes under testningen mer ingående, eller att identifiera ytterligare hot som inte framkom i den initiala genomgången. I det här fallet prioriterades dock en bred översikt av enheternas yttre säkerhet för att bedöma den generella nivån av sårbarhet, snarare än att fokusera på enskilda attackytor i detalj.

Port 5355, som används av LLMNR (Link-Local Multicast Name Resolution), identifierades som öppen i det isolerade laborationsnätverket. Även om detta i labbmiljön inte medförde en omedelbar säkerhetsrisk, eftersom nätverket saknade extern åtkomst, utgör förekomsten av LLMNR i sig en potentiell sårbarhet. Protokollet saknar autentiseringsmekanismer och är känt för att kunna utnyttjas i spoofingattacker för att fånga upp autentiseringsuppgifter. Att tjänsten var aktiverad antyder brist på hårdare segmentering eller grundläggande härdning, vilket i en produktionsmiljö skulle klassas som en säkerhetsbrist. Detta understryker vikten av att även i testmiljöer reflektera över vilka tjänster som exponeras, särskilt om dessa kan leda till felaktiga konfigurationer i skarpa driftsmiljöer (Microsoft 2023).

Vidare observerades att port 1534 (micromuse-lm) var öppen, men att tjänsten som använde den inte kunde identifieras. Då porten i normalfallet inte är associerad med väldokumenterade offentliga tjänster, utan snarare används av proprietära lösningar, var det inte möjligt att genomföra djupare tester utan mer information. Från ett säkerhetsperspektiv innebär detta att potentiella sårbarheter kan förbli oupptäckta, vilket betonar vikten av att organisationer har kontroll över alla exponerade portar – särskilt de som inte är allmänt erkända eller dokumenterade. Begränsningar i tid och omfattning medförde dock att analysen av denna port inte prioriterades i denna studie (IANA 2025).

5.2 Resultatets koppling till direktiven

Resultaten från penetrationstesterna kopplas direkt till kraven i de olika direktiven och används för att bedöma om de testade enheterna uppfyller dessa säkerhetskrav. Ett tydligt mönster som observerades var att de flesta enheterna hade stängt av alla onödiga portar, och endast lämnat de nödvändiga öppna. Detta är i linje med flera av de aktuella direktiven, särskilt RED-DA, som ställer krav på att produkter ska skyddas mot obehörig åtkomst. Om en enhet har många öppna portar som inte fyller någon funktion, ökar risken för att en angripare kan utnyttja dem.

När en angripare väl får tillgång till en öppen port som inte är tillräckligt skyddad, kan det fungera som en inkörsport till vidare kompromettering av systemet. Detta kan möjliggöra allt från installation av skadlig kod till obehörig åtkomst av känsliga data eller kontroll över enhetens funktioner. I vissa fall kan det även leda till lateral rörelse i nätverket, där angriparen sprider sig till andra enheter eller system. Därför betraktas ett strikt begränsat antal öppna portar, kombinerat med korrekt konfiguration och övervakning, som en grundläggande och kritisk säkerhetsåtgärd. Detta är inte bara en teknisk rekommendation, utan också ett tydligt krav i säkerhetsstandarder och EU-direktiv som RED-DA och CRA, vilka betonar vikten av att minimera exponerade angreppsytor för att förebygga obehörig åtkomst och säkerställa produktsäkerhet under hela livscykeln (Pagnotta 2024).

Vidare upptäcktes på vissa enheter möjligheten till obegränsade inloggningsförsök utan några former av skyddsmekanismer, såsom kontolåsning eller CAPTCHA. Detta innebär att en angripare potentiellt skulle kunna genomföra en så kallad brute-force-attack utan att stoppas. Detta hot faller under både RED-DA och CRA, då sådana sårbarheter direkt påverkar integritet, autentisering och skydd mot obehörig åtkomst, vilket är kärnaspekter i direktivens krav på grundläggande cybersäkerhet. Detta är särskilt allvarligt eftersom det innebär att en angripare kan genomföra ett obegränsat antal inloggningsförsök utan att möta några motåtgärder, såsom kontolåsning, tidsfördröjningar eller CAPTCHA-skydd. Utan sådana spärrar öppnas dörren för automatiserade brute-force-attacker, där ett skript systematiskt kan prova miljontals lösenord inom kort tid tills rätt kombination hittas. Denna typ av angrepp kräver varken avancerade verktyg eller djup teknisk kunskap, vilket gör dem särskilt attraktiva för illasinnade aktörer. Att sakna skydd mot detta underminerar helt grundläggande säkerhetsprinciper som konfidentialitet och autentisering, båda centrala delar av de regulatoriska krav som ställs i RED-DA och CRA (Esheridan u.å).

Det är dock viktigt att påpeka att kontolåsning efter ett visst antal misslyckade inloggningsförsök också kan medföra säkerhetsrisker. En angripare kan utnyttja denna funktion för att medvetet låsa ut legitima användare genom upprepade felaktiga inloggningsförsök, vilket skapar en form av DoS-attack. Utöver detta kan funktionen oavsiktligt läcka information om vilka konton som existerar i systemet, om ett konto endast kan spärras om det faktiskt finns, kan en angripare dra slutsatsen att ett spärrat konto är giltigt. Ett mer balanserat skydd är att implementera IP-baserad blockering, vilket minskar risken för både kontoutlåsning av legitima användare och kontoinformationsexponering. Enhet E är ett gott exempel på detta, då den hade ett sådant skydd implementerat. Genom att blockera misstänkt trafik baserat på IP-adresser istället för att låsa själva kontot, kan man effektivt försvåra brute-force-attacker utan att påverka den legitima användarens åtkomst (Esheridan u.å).

Under testerna visade det sig att en av de allvarligaste sårbarheterna hos vissa enheter var deras bristande motståndskraft mot DoS-attacker. Flera enheter påverkades så kraftigt att de helt slutade fungera under tiden som attacken pågick, vilket i praktiken gjorde dem otillgängliga. Andra enheter visade viss motståndskraft men uppvisade tydliga prestandaproblem, såsom extremt långsamma svarstider eller kraftigt försämrad funktionalitet.

Denna typ av sårbarhet är direkt kopplad till principerna i både CRA och NIS2-direktivet. Enligt CRA måste digitala produkter, inklusive IoT-enheter, vara konstruerade för att motstå vanliga typer av attacker, däribland överbelastningsattacker, under hela sin livscykel. Om en enhet lätt kan slås ut av en enkel DoS-attack indikerar det brister i dess grundläggande säkerhet och motståndskraft, vilket går emot direktivets krav på säkerhet som standard.

Utöver dessa aspekter är det också relevant att koppla resultaten till mer specifika krav i RED DA-direktivet. Enligt punkt d i RED DA ska radioutrustning vara konstruerad på ett sätt som inte skadar nätverket, dess funktion eller orsakar oacceptabel försämring av tjänsten. De enheter som under DoS-attacker slutade fungera eller visade tydliga tecken på prestandadegradering kan därmed anses brista i detta avseende. En produkt som lätt kan slås ut påverkar inte bara den egna funktionen utan även tillgängligheten och stabiliteten i det nätverk där den är installerad, vilket gör detta till ett allvarligt avsteg från direktivets intentioner enheter *Direktiv 2014/53/EU om harmonisering av medlemsstaternas lagstiftning om tillhandahållande av radioutrustning på marknaden.*

Det är därför avgörande att kunna mitigera DoS-attacker på ett effektivt sätt för att undvika omfattande skador på system och tjänster. En central del i denna process är förmågan att först detektera skadlig trafik. Under detektionsfasen är det särskilt viktigt att kunna skilja mellan legitim och skadlig trafik, vilket utgör en stor utmaning inom cybersäkerhet (Mahjabin, Xiao, Sun & Jiang 2017).

Det finns flera metoder för att identifiera skadlig trafik, varav två vanliga är signature-based detection och anomaly-based detection. Signature-based detection bygger på igenkänning av tidigare kända attacker genom definierade mönster eller "signaturer". Fördelen med denna metod är att den är snabb och effektiv vid upptäckt av välkända attacker, men dess begränsning ligger i att den har svårt att upptäcka nya eller okända hot, såsom så kallade zero-day-attacker, där sårbarheter ännu inte är dokumenterade (Mahjabin et al. 2017).

Anomaly-based detection, fokuserar på att identifiera avvikelser från normalt beteende i nätverkstrafiken och har därmed bättre möjligheter att upptäcka just nya och okända attacker. Nackdelen med denna metod är dock att den är mer benägen att generera falska alarm, alltså situationer där legitim trafik felaktigt tolkas som skadlig, vilket kan leda till onödiga insatser eller driftstörningar (Mahjabin et al. 2017).

När en DoS-attack har detekterats är det avgörande att snabbt vidta åtgärder för att hantera situationen. En effektiv respons kan vara avgörande, inte bara för att minimera attackens konsekvenser, utan i vissa fall även för att helt eliminera hotet. Det finns flera metoder för att bemöta en pågående DoS-attack. Två vanliga tekniker är filtrering och rate limiting. Om det är möjligt att tydligt särskilja skadlig trafik från legitim trafik är filtrering att föredra, då den möjliggör att endast den skadliga trafiken blockeras utan att störa normal användning. Om skillnaden däremot är otydlig eller svår att avgöra i realtid, kan rate limiting vara en mer lämplig strategi. Genom att begränsa mängden trafik som tillåts från en viss källa inom ett visst tidsintervall kan man effektivt minska påverkan från potentiellt skadlig trafik, samtidigt som viss tillgång för legitim trafik bibehålls (Mahjabin et al. 2017).

Sårbarheten mot DoS-attacker är då av särskild betydelse inom ramen för NIS2, särskilt om enheten ingår i verksamheter som klassas som samhällskritiska. NIS2 ställer krav på att sådana verksamheter ska kunna säkerställa tillgänglighet, ett av de tre grundelementen i informationssäkerhet (tillsammans med konfidentialitet och integritet). En enhet som kan sättas ur spel genom en relativt enkel attack riskerar att skapa driftavbrott eller tillgänglighetsproblem, något som enligt direktivet måste förebyggas och åtgärdas med hjälp av både tekniska och organisatoriska skyddsåtgärder.

Vidare bör punkt e uppmärksammas, som kräver att utrustning som hanterar personuppgifter eller lokaliseringsdata ska ha inbyggda funktioner för att skydda dessa uppgifter. Även om detta inte var fokus för samtliga tester, visade flera enheter bristande skydd mot obehörig åtkomst via svaga autentiseringsmekanismer. Detta ökar risken för dataintrång och därmed för att känslig information exponeras. Sådana brister strider mot kraven på integritetsskydd som uttrycks i både RED DA och CRA enheter *Direktiv 2014/53/EU om harmonisering av medlemsstaternas lagstiftning om tillhandahållande av radioutrustning på marknaden.*

Punkt f, som rör skydd mot bedrägerier vid överföring av penningvärden eller digitala valutor, var inte direkt tillämplig för de testade enheterna, då ingen av dem hade funktioner för betalningshantering. Den bör dock beaktas vid granskning av mer avancerad radioutrustning eller konsumentenheter som erbjuder transaktionsmöjligheter *Direktiv 2014/53/EU om*

harmonisering av medlemsstaternas lagstiftning om tillhandahållande av radioutrustning på marknaden.

5.3 Riskbedömning Med DREAD-Modellen

För att systematiskt bedöma de identifierade hoten under penetrationstesterna tillämpades DREAD-modellen, vilken utvärderar varje risk utifrån fem faktorer: Damage Potential (D), Reproducibility (R), Exploitability (E), Affected Users (A), samt Discoverability (D). Nedan följer en sammanställning per enhet, med tillhörande analys. Skalan för poängsättning inom respektive kategori definierades i avsnitt 3.5 och har använts konsekvent i nedanstående bedömningar.

Tabell 1 Riskbedömning för Enhet A enligt DREAD-Modellen

Hot	D	R	E	A	D
DOS via ICMP/SYN flood	10	10	5	10	5
Brute-force av dolda katagorier	5	7.5	5	2.5	8
CVE	0	10	10	0	10
Websocket / injektionsförsök	5	7.5	5	2.5	5

Den största risken för enhet A är DoS-attacker (ICMP och SYN flood), eftersom de kan påverka tillgängligheten för alla användare och är enkla att reproducera med vanliga verktyg. Övriga sårbarheter är i nuläget informationsbaserade eller potentiella, men bör följas upp med härdning och förbättrad loggning. Det är även viktigt att implementera skydd mot brute-force och förbättra filtreringen av exponering mot klienten.

Tabell 2 Riskbedömning för Enhet B enligt DREAD-Modellen

Hot	D	R	E	A	D
DoS via RPCBOMB (port 111)	8	5	5	10	5
DoS via ICMP/SYN flood	8	7.5	5	10	5
Brute-force via SSH (port 22)	5	5	2.5	2.5	5
Informationsläckage via RpcInfo	8	5	5	8	5
CVE sårbarheter i Linux 5.8	5	10	10	10	8

Enhet B uppvisar hög risk för överbelastningsattacker, särskilt genom SYN-flood och RPCBOMB. Systemet är även potentiellt sårbart mot SSH-bruteforce, även om inga intrång

lyckades i detta fall. Att RPC-tjänster är exponerade och identifierbara bör åtgärdas, liksom en uppdatering eller härdning av Linux-versionen om några kritiska CVE:er matchar systemet.

Tabell 3Riskbedömning för Enhet C enligt DREAD-Modellen

Hot	D	R	E	A	D
Brute-force utan kontolåsning	8	5	5	2.5	5
Tjänsteavbrott/DoS via katalogskanning	9	5	5	10	0
Informationsläckage via serversignatur	5	10	10	10	10

Enhet C uppvisar risk för tillgänglighetsangrepp då systemet kraschade vid katalogskanning med Wfuzz, vilket tyder på bristande felhantering. Avsaknaden av spärrmekanismer vid upprepade inloggningsförsök öppnar även för brute-force-attacker. Att Apache-versionen avslöjas innebär en potentiell sårbarhet om tjänsten inte är uppdaterad. Åtgärder bör vidtas för att förbättra både robusthet och informationshantering.

Tabell 4 Riskbedömning för Enhet D enligt DREAD-Modellen

Hot	D	R	E	A	D
Brute-force utan kontolåsning	8	5	5	2.5	5
DoS via SYN-flood	8	7.5	5	10	5
Information läckage via SMTP-banner	8	5	5	6	5
Robots-txt och root-användaren via Wfuzz	5	7.5	5	8	5

Enhet D har hög risk kopplad till brist på brute-force-skydd samt sårbarhet för DoS genom SYN-floods. Att användarnamnet @root kunde identifieras samt att robots.txt ger extra information ökar riskexponeringen för riktade attacker. SMTP-banner ger insyn i tjänsteversion men kräver kompletterande steg för att bli farlig.

Tabell 5 Riskbedömning för Enhet E enligt DREAD-Modellen

Hot	D	R	E	A	D
SYN-flood	8	7.5	5	10	5
ICMP-flood	5	7.5	5	10	5
MQTT DoS med python-script	5	5	2.5	10	8
Dolda kataloger & hash-analyser (Gobuster, favicon)	5	7.5	5	2.5	5
Brute-force på inloggningssidan	2	5	5	2.5	5

Enhet E uppvisar störst sårbarhet mot nätverksburna överbelastningsattacker, särskilt SYN-floods som orsakar total nedgång i tjänsten. Även MQTT-tjänsten på port 1883 är potentiellt sårbar om tillgång finns via externa nätverk. Positivt är att inloggningsmekanismen har ett effektivt brute-force-skydd per IP, vilket kraftigt reducerar risken för autentiseringsangrepp.

Tabell 6 Riskbedömning för Enhet F enligt DREAD-Modellen

Hot	D	R	E	A	D
Brute-force mot SSH	5	5	2.5	2.5	5
DoS via SYN-flood	5	7.5	5	10	5
DoS via UDP-flood	5	7.5	5	10	5
DoS via ICMP-flood	5	7.5	5	10	5

Enhet F är särskilt sårbar för brute-force attacker mot SSH, eftersom port 22 är öppen och inga motåtgärder upptäcktes i testet. Även SYN-floods mot port 2033 kan påverka tillgängligheten kraftigt. Positivt är att ICMP-floods inte påverkade tillgängligheten, och att enhetens exponering för CVE-exploateringar är låg så länge uppdateringar tillämpas och tjänster hålls säkra.

Tabell 7 Riskbedömning för Enhet G enligt DREAD-Modellen

Hot	D	R	E	A	D
SYN-flood	5	7.5	5	10	5
ICMP-flood	5	7.5	5	10	5
UDP-flood	5	7.5	5	10	5

Enhet G har en relativt bra skydd mot överbelastningsattacker, då inga prestandanedsättningar observerades vid DoS-testning. De främsta riskerna ligger i exponeringen av flera portar och potentiella CVE-sårbarheter kopplade till identifierade komponenter från ”Företag AB”. Dessa bör analyseras djupare för att säkerställa att uppdateringar och patchar är tillämpade.

6. DISKUSSION

Syftet med denna studie har varit att undersöka om de testade enheterna uppfyller säkerhetskraven enligt aktuella EU-direktiv, däribland NIS2, RED-DA och CRA. Genom en kombination av teoretisk genomgång och praktiska penetrationstester har vi kunnat identifiera både styrkor och svagheter i enheternas säkerhetsimplementationer.

6.1 Tolkning av resultat

Generellt visar resultaten att flera av enheterna uppvisar ett grundläggande skydd mot vanliga hot, såsom stängda portar och visst skydd mot brute-force-attacker. Detta överensstämmer med de krav på ”security by design” som återfinns i RED-DA och CRA. Särskilt positivt är att många enheter hade minimerat antalet öppna portar, vilket reducerar angreppsytan och därmed risken för obehörig åtkomst.

Dock identifierades också ett antal allvarliga brister, framför allt kopplade till sårbarhet för överbelastningsattacker. Flera enheter blev obrukbara under relativt enkla DoS-tester, något som strider mot CRA:s krav på motståndskraft mot vanliga attacker samt NIS2:s fokus på tillgänglighet i samhällskritiska system. Dessa resultat tyder på att trots goda grundläggande skydd finns det fortfarande en betydande brist i robusthet mot driftstörningar.

Ett annat återkommande problem var avsaknaden av skydd mot obegränsade inloggningsförsök. Detta ökar risken för lyckade brute-force-attacker, och visar på brister i autentiseringsskydd, vilket är centralt i såväl RED-DA som CRA (Ruohonenra 2024).

6.2 Metodens begränsningar

Det är viktigt att reflektera över att penetrationstesterna utfördes under begränsad tid och med begränsade resurser. Vår tekniska förförståelse var i viss mån otillräcklig, vilket gjorde att en betydande del av arbetet gick åt till att söka information och sätta upp verktyg snarare än att genomföra djupgående attacker. Detta påverkade testernas omfattning och djup, och kan innebära att vissa sårbarheter förblev oupptäckta. Samtidigt understryker det en central poäng: att genomföra säkerhetstester kräver noggranna förberedelser och rätt kompetens, vilket i sig är en utmaning för många organisationer som vill uppfylla direktivens krav.

Att vi inte gick vidare med port 2033, trots att den var öppen, är ett exempel på detta. Eftersom studien omfattar grundläggande penetrationstester av flera olika enheter, har fokus legat på att identifiera generella säkerhetsbrister snarare än att genomföra djupgående analyser av enskilda tjänster. Detta innebär en metodologisk begränsning, men speglar samtidigt verkligheten för många IT-säkerhetsansvariga: det är ofta nödvändigt att prioritera bredd framför djup på grund av begränsade resurser och tid.

6.3 Relevans i ett större sammanhang

Våra resultat visar att även moderna nätverksanslutna enheter kan ha kritiska sårbarheter, särskilt när det gäller tillgänglighet och skydd mot automatiserade angrepp. I takt med att EU:s direktiv blir bindande ställs högre krav på tillverkare att inte bara säkra sina produkter vid lansering, utan att aktivt arbeta med sårbarhetsreducering under hela livscykeln. Våra tester illustrerar hur viktigt det är att dessa krav konkretiseras och implementeras, inte bara i teori, utan i faktisk produktdesign och leverans.

6.4 Framtida arbete

Utifrån studiens genomförande finns det flera tekniska aspekter som skulle kunna förbättras i ett framtida projekt med liknande upplägg. En mer detaljerad testplan med tydliga procedurer för varje enhet skulle kunna säkerställa att fler sårbarheter identifieras konsekvent. Det vore även fördelaktigt att inkludera ett större inslag av automatiserade verktyg tidigt i processen för att effektivisera testningen och frigöra tid för djupare manuell analys.

Som tidigare nämnts har den främsta prioriteringen i denna studie varit att genomföra grundläggande penetrationstester på samtliga enheter. Detta angreppssätt innebar att vissa öppna portar identifierades men inte undersöktes vidare, trots att de potentiellt hade kunnat avslöja ytterligare sårbarheter vid en mer fördjupad analys. Mer detaljerade tester av dessa portar hade exempelvis kunnat avslöja osäkra tjänstekonfigurationer, bristande autentiseringsrutiner eller möjligheter till obehörig åtkomst via svaga protokoll. Eftersom varje öppen port kan exponera en unik attackyta, innebär ett djupare utforskande ofta att mer komplexa eller mindre uppenbara sårbarheter upptäcks, sårbarheter som kan vara avgörande för en angripares möjlighet att ta sig vidare i ett system. Därför utgör detta ett naturligt fokusområde för framtida arbete, där testningen kan kompletteras med större teknisk djupgående analys per enhet.

Vidare bör verktyg och tekniker som visat sig effektiva i denna studie, exempelvis DoS-simulering, formaliseras och dokumenteras för att återanvändas i framtida tester. Att i ett senare skede inkludera eftertester (till exempel för att se hur enheterna återhämtar sig efter en attack) skulle också ge en mer komplett bild av säkerhetsnivån.

Utöver det tekniska angreppssättet vore det värdefullt att även analysera tillverkarnas hantering av sårbarheter efter produktlansering. CRA-direktivet ställer tydliga krav på kontinuerlig uppdatering och patchhantering, och framtida studier skulle därför kunna undersöka hur snabbt och effektivt uppdateringar erbjuds efter upptäckta brister. Detta skulle ge insikt i den långsiktiga cybersäkerheten och efterlevnaden av direktivens livscykelperspektiv.

Ett ytterligare steg vore att genomföra tester i mer verklighetstrogna miljöer, exempelvis simulerade produktionsnätverk, för att bättre förstå de praktiska konsekvenserna av sårbarheter. Detta skulle kunna kombineras med hotmodellering (threat modeling) som förberedande moment, där sannolika attackvägar och särskilt utsatta komponenter identifieras redan innan testningen påbörjas.

Slutligen kan framtida projekt även inkludera jämförande analyser mellan produkter från olika tillverkare för att identifiera gemensamma mönster, återkommande brister eller framgångsrika skyddsmekanismer. Sådan kunskap kan vara värdefull inte bara för tillverkare och säkerhetsansvariga, utan även för lagstiftare och standardiseringsorgan i deras arbete med att utveckla effektiva cybersäkerhetskrav.

6.5 Brister i efterlevnad

De identifierade sårbarheterna i de testade enheterna väcker frågan varför dessa brister kvarstår, trots tydliga krav i EU:s regelverk. En möjlig förklaring är den tidsförskjutning som ofta uppstår mellan lagstiftningens införande och faktisk implementering hos tillverkare. Flera enheter kan ha utvecklats innan de aktuella direktiven trädde i kraft eller konkretiserades i

tekniska standarder. Detta gäller särskilt CRA, där krav på cybersäkerhet i hela produktlivscykeln ställer höga krav på utvecklingsprocesser, underhåll och dokumentation.

En annan viktig faktor är bristen på incitament och tillsyn. Små eller medelstora tillverkare, särskilt utanför EU, kan sakna resurser för att genomföra systematiska säkerhetstester eller för att uppdatera produkter i linje med direktiven. Därtill försvåras efterlevnaden av komplexa leverantörskedjor, där komponenter och programvara från tredjepartsleverantörer kan introducera sårbarheter som huvudtillverkaren har begränsad kontroll över. Detta belyser vikten av att säkerhetsgranska hela produktens ekosystem.

Resultatet visar också att vissa brister, såsom avsaknad av skydd mot brute-force-attacker kan bero på bristande kunskap eller rutiner i utvecklingsfasen. För att adressera detta krävs att tillverkare implementerar säkerhetsprinciper som secure by default, rate limiting, tydliga autentiseringsflöden samt rutiner för sårbarhetsrapportering och regelbundna uppdateringar. Direktivens krav på ”security by design” måste därmed omsättas i praktiska och tekniska åtgärder genom hela utvecklingskedjan. För att detta ska bli verklighet behöver cybersäkerhet bli en integrerad del av hela produktutvecklingen och inte ett tillägg i slutet.

7. SLUTSATSER OCH VIDARE FORSKNING

I detta kapitel sammanfattas studiens huvudsakliga slutsatser utifrån de formulerade forskningsfrågorna och resultaten från genomförda penetrationstester. Studien syftade till att undersöka i vilken utsträckning ett urval av nätverksanslutna enheter uppfyller säkerhetskraven i de tre aktuella EU-direktiven: NIS2, RED DA och CRA. Slutsatserna nedan är strukturerade utifrån studiens tre forskningsfrågor, som fungerade som vägledande ramverk genom hela undersökningen.

7.1 Slutsats

För att uppnå syftet har följande forskningsfrågor besvarats:

- Hur väl uppfyller de testade enheterna säkerhetskraven enligt NIS2-, RED DA- och CRA-direktiven?
- Vilka sårbarheter kan identifieras genom penetrationstester av de utvalda enheterna?
- Vilken inverkan har de identifierade sårbarheterna på enheternas förmåga att uppfylla EU:s cybersäkerhetskrav?

Resultaten från penetrationstesterna visar tydliga kopplingar mellan de identifierade sårbarheterna och de krav som ställs i direktiven. Många av enheterna visade brister i grundläggande cybersäkerhet, exempelvis genom att exponera onödiga öppna portar, sakna skydd mot brute-force-attacker eller vara känsliga för DoS-attacker. Dessa brister utgör konkreta risker som påverkar både tillgänglighet, dataintegritet och autentisering, vilket är centrala principer i direktiven.

Särskilt framträdande var effekterna av DoS-attacker, där flera enheter helt eller delvis slogs ut under angreppet. Detta står i strid med kraven på tillgänglighet och robusthet i samtliga tre direktiv, särskilt inom NIS2 som betonar driftsäkerhet i samhällskritiska funktioner. Även RED-DA:s krav på skydd mot obehörig åtkomst, och CRA:s betoning på säkerhet genom hela livscykeln, sätts i spel när grundläggande säkerhetsåtgärder saknas eller brister.

Sammanfattningsvis visar analysen att flera av de testade enheterna inte fullt ut uppfyller EU:s cybersäkerhetskrav. Frågeställningarna kan därför besvaras enligt följande:

- De testade enheterna uppfyller endast delvis kraven enligt NIS2-, RED DA- och CRA-direktiven.
- Identifierade sårbarheter omfattade otillräckligt skyddade nätverksportar, avsaknad av inloggningsskydd samt låg motståndskraft mot överbelastningsattacker.
- Dessa sårbarheter försämrar enheternas förmåga att uppfylla direktivens krav på säkerhet, tillgänglighet, integritet och skydd mot obehörig åtkomst.

Penetrationstester visade sig därmed vara ett effektivt verktyg för att bedöma cybersäkerheten i praktiken, och resultaten betonar behovet av förbättrade säkerhetsåtgärder för att minska sårbarheter och uppnå regelmässig efterlevnad

7.2 Framtida forskning

Eftersom denna studie endast omfattade ett begränsat antal enheter och genomfördes under en begränsad tidsram, finns det flera möjligheter att vidareutveckla forskningen inom detta område. Framtida studier bör inkludera ett bredare urval av nätverksanslutna enheter från olika produktkategorier och tillverkare för att möjliggöra mer generaliserbara slutsatser

gällande efterlevnaden av NIS2-, RED DA- och CRA-direktiven. Det vore även värdefullt att standardisera testmiljön genom att använda en enhetlig uppsättning testverktyg och metoder för liknande typer av enheter, vilket skulle underlätta jämförelser och tydliggöra var sårbarheter respektive styrkor föreligger.

Vidare skulle en mer omfattande användning av automatiserade penetrationstester, i kombination med djupgående sårbarhetsanalyser, kunna ge en mer detaljerad bild av enheternas faktiska säkerhetsnivå. Ett annat relevant forskningsområde är att undersöka hur tillverkare arbetar med att åtgärda identifierade sårbarheter, implementera säkerhetsförbättringar och förebygga återkommande brister.

Slutligen bör framtida forskning även utforska möjligheterna till ökat samarbete mellan forskare, tillverkare och branschaktörer. Ett sådant tvärsektoriellt samarbete kan bidra till utvecklingen av mer robusta cybersäkerhetsstrategier samt främja en ökad förståelse för de risker som är förknippade med uppkopplade IoT- och OT-enheter i både hem- och industrimiljöer.

8. REFERENSER

Aibekova, A. & Selvarajah, V. (2022) Offensive Security: Study on Penetration Testing Attacks, Methods, and their Types, *2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE).* Ballari, Indien 23-24 april 2022, s. 1-9. https://doi.org/10.1109/ICDCECE53908.2022.9792772

America's cyber defense agency (CISA) (u.å) *Nikto*
https://www.cisa.gov/resources-tools/services/nikto [2025-03-15]

Antonakakis, M. April, T. Bailey, M. Bernhard, M. Bursztein, E. Cochran, J. Halderman, A. Invernizzi, L. Kumar, D. Lever, C. Ma, Z. Mason, J. Menscher, D. Sullivan, N. Thomas, K. & Zhou, Y. (2017). Understanding the Mirai Botnet. *This paper is included in the Proceedings of the 26th USENIX Security Symposium*. Vancouver, Kanada 16-18 augusti 2017, s 1-19. https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis

Bakry, B. Adenan, A. & Yussoff, Y. (2022). Security Attack on IoT Related Devices Using Raspberry Pi and Kali Linux, *2022 International Conference on Computer and Drone Applications (IConDA)*, Kuching, Malaysia, 28-29 november 2022, s. 40-45. https://doi.org/10.1109/ICONDA56696.2022.10000370

Bhole, M. Kastner, W. & Sauter, T. (2024). IT Security Solutions for IT/OT Integration: Identifying Gaps and Opportunities, *2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA).* Padova, Italien 10-13 september 2024, s. 1-8. https://doi.org/10.1109/ETFA61755.2024.10710968

Chiara, P. G. (2022) *The IoT and the new EU cybersecurity regulatory landscape*, International Review of Law, Computers & Technology, 36(2). https://doi.org/10.1080/13600869.2022.2060468

Cloudflare (u.å.). *Ping (ICMP) flood DDoS attack*.
https://www.cloudflare.com/learning/ddos/ping-icmp-flood-ddos-attack [2025-03-10].

Covert, J., Francis, D., Steinhagen, P. & Streff, K. (2020). *CIA Triad: A Security Framework*. Proceedings of the 53rd Hawaii International Conference on System Sciences. https://scholarspace.manoa.hawaii.edu/server/api/core/bitstreams/5486a250-cc3c-4227-a752-7d08378afbdf/content

CVE (u.å). *About the CVE Program*
https://www.cve.org/About/Overview [2025-02-10]

Direktiv 2014/53/EU om harmonisering av medlemsstaternas lagstiftning om tillhandahållande av radioutrustning på marknaden. (EUT L 153, 22.5.2014, s. 62–106). https://eur-lex.europa.eu/legal-content/SV/TXT/?uri=CELEX%3A32014L0053

Eckhardt, P. & Kotovskaia, A. (2023). *The EU's cybersecurity framework: the interplay between the Cyber Resilience Act and the NIS 2 Directive*. International Cybersecurity Law Review.
https://doi.org/10.1365/s43439-023-00084-z

Esheridan (u.å.). *Blocking Brute Force Attacks*.
https://owasp.org/www-community/controls/Blocking_Brute_Force_Attacks [2024-10-22].

European Cyber Resilience Act (u.å.). *Cyber Resilience Act (CRA) | Updates, Compliance*
https://www.european-cyber-resilience-act.com/ [2025-02-13]

Falch, M. Henten, A.H. Kaloudis, A & Windekiælde, I. (2024). *Economics and regulation of cybersecurity - the case of E.* SSRN, pp. 1-18.
https://vbn.aau.dk/ws/portalfiles/portal/756599111/Paper_on_cybersecurity_economics_and_regulation_preprint.pdf

GeeksforGeeks (2023). *Introduction to Netcat*.
https://www.geeksforgeeks.org/introduction-to-netcat/ [2025-03-11]

Goel, J. N. Mehtre, B. M. (2015). *Vulnerability Assessment & Penetration Testing as a Cyber Defence Technology*, Procedia Computer Science
https://doi.org/10.1016/j.procs.2015.07.458

Greenbone Networks (u.å.). *Greenbone Networks - Open Source Vulnerability Management*.
https://www.greenbone.net/en/ [2025-03-11]

Gustafsson, D. (@halvtomat). (2022). *swedish-passwords* [githubinlägg], 13 juni.
https://github.com/halvtomat [2025-03-12]

IANA (2025). *Service Name and Transport Protocol Port Number Registry*.
https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml [2025-03-11]

Jara, A. Martinez, I. & Sanchez, J. (2024). CyberSecurity Resilience Act (CRA) in Practice for IoT Devices: Getting Ready for the NIS2, *2024 IEEE Smart Cities Futures Summit (SCFC),* Marrakech, Marocko 29-31 maj 2024, s. 56-60.
https://doi.org/10.1109/SCFC62024.2024.10698057

Johari, K. Karthikram, A. Kumar, K. Reddy, S. & Kumar, P. Automated Web Application Hacking Framework. (2024) *International Conference on Computing and Data Science (ICCDS)*. Chennai, Indien 26-27 april 2024, s. 1-6.
https://doi.org/10.1109/ICCDS60734.2024.10560380

Kali Linux (2024). *hping3*.
https://www.kali.org/tools/hping3/ [2025-03-11]

Kim, J. (2020). *Burp Suite: Automating Web Vulnerability Scanning*. Masteruppsats, ProQuest Dissertations & Theses. Utica College.
https://www.proquest.com/openview/cdef16944ee151614dd38c0e09b24c13/1?cbl=18750&diss=y&pq-origsite=gscholar

Kim, K. H. Kim, K. Kim, H. K. (2022). *STRIDE-based threat modeling and DREAD evaluation for the distributed control system in the oil refinery*. Wiley Online Library.
https://doi.org/10.4218/etrij.2021-0181

Kommissionens delegerade förordning (EU) 2022/30 av den 29 oktober 2021 om komplettering av direktiv 2014/53/EU vad gäller tillämpningen av de väsentliga krav som

avses i artikel 3.3 d, e och f i det direktivet (Text av betydelse för EES). (EUT L 7, 12.1.2022, s. 6–10). https://eur-lex.europa.eu/legal-content/SV/ALL/?uri=CELEX%3A32022R0030

Lockheed Martin (u.å.). *Cyber Kill Chain*.
https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html [2025-02-20]

Mahjabin T, Xiao Y, Sun G, Jiang W. (2017). *A survey of distributed denial-of-service attack, prevention, and mitigation techniques.* International Journal of Distributed Sensor Networks. https://doi.org/10.1177/1550147717741463

Microsoft (2023) *Microsoft Security Bulletin MS11-030 - Critical*
https://learn.microsoft.com/en-us/security-updates/securitybulletins/2011/ms11-030 [2025-04-12sof]

MITRE (u.å.). *ATT&CK Matrix for Enterprise*.
https://attack.mitre.org/ [2025-02-26]

National cyber security centre. (2023). *Risk management.*
https://www.ncsc.gov.uk/collection/risk-management/threat-modelling [2025-02-24]

Nmap (u.å.). *Nmap*.
https://nmap.org/ [2025-03-10]

Oberheide, J. (2020). *Practical Web Application Security*. Sebastopol.
https://learning.oreilly.com/library/view/web-application-security/9781492053101/

Penetration Testing Execution Standard (PTES). (2014). *High Level Organization of the Standard.*
http://www.pentest-standard.org/index.php/Main_Page [2025-03-03]

Pagnotta, S. (2024). Open Ports: Are they a Vulnerability? *Bitsight* [blogg], 16 februari.
https://www.bitsight.com/blog/open-port-vulnerabilities-whats-the-big-deal [2025-04-12]

Yuga (2023). DREAD Threat Modeling Methodology *Practical Devsecops* [blogg], 9 mars.
https://www.practical-devsecops.com/dread-threat-modeling/ [2025-02-26]

Raj, S. & Walia, N. A Study on Metasploit Framework: A Pen-Testing Tool. (2020). *International Conference on Computational Performance Evaluation (ComPE)*. Shillong, Indien 2-4 juli 2020, s. 296-302, https://doi.org/10.1109/ComPE49325.2020.9200028

Rashid, M.M. (2023). *Qualitative Research and Quantitative Research.* Conference paper.
https://doi.org/10.13140/RG.2.2.35111.05280

Ruohonenra, J. (2024) *A Systematic Literature Review on the NIS2 Directive*
https://arxiv.org/pdf/2412.08084

Schuba, C. L. Krsul, I. V. Kuhn, M. G. Spafford, E. H. Sundaram, A. & Zamboni, D. (1997) "*Analysis of a denial of service attack on TCP*," Proceedings. 1997 IEEE Symposium on Security and Privacy (Cat. No.97CB36097)
https://doi.org/10.1109/SECPRI.1997.601338

Snider, K. Shandler, R. Zandani, S. & Canetti, D. *Cyberattacks, cyber threats, and attitudes toward cybersecurity policies*, Journal of Cybersecurity. https://doi.org/10.1093/cybsec/tyab019

Soepeno, R. (2023). *Wireshark: An Effective Tool for Network Analysis.* ResearchGate. https://doi.org/10.13140/RG.2.2.34444.69769

Singh, C. (2023). The European Approach to Cybersecurity in 2023: A Review of the Changes Brought in By the Network and Information Security 2 (NIS2) Directive 2022/2555. *International Company and Commercial Law Review.* (5), s. 251–261. https://westminsterresearch.westminster.ac.uk/download/96f4af83e310a7f01c883b9db11375f4e378b0fd1681e82bb288d3d783bc98ae/303407/Information%20Security%202%20%28NIS2%29%20Directive%202022%202555.pdf

Vaishnavi. (2024). How Ethical Hackers Leverage Netcat for Network Debugging: Overview, Features, and Why Ethical Hackers Use It. *WebAsha Technologies* [blogg], 23 december. https://www.webasha.com/blog/how-ethical-hackers-leverage-netcat-for-network-debugging-overview-features-and-why-ethical-hackers-use-it [2025-03-10]

Vandezande, N. (2024) Cybersecurity in the EU: How the NIS2-directive stacks up against its predecessor. *Computer Law & Security Review,* 52, Artikel 105890. https://doi.org/10.1016/j.clsr.2023.105890

Vugdelija, N. Nedeljković, N. Kojić, N. Lukić, L. & Vesić, M. (2021). *Review of Brute-Force Attack and Protection Techniques* https://proceedings.ictinnovations.org/attachment/paper/554/review-of-brute-force-attack-and-protection-techniques.pdf

Wfuzz (u.å). *Wfuzz: The Web fuzzer*. https://wfuzz.readthedocs.io/en/latest/ [2025-03-11]

HÖGSKOLAN
I BORÅS

Besöksadress: Allégatan 1 · Postadress: 501 90 Borås · Tfn: 033-435 40 00 · E-post: registrator@hb.se · Webb: www.hb.se