Definizione di un framework per la verifica della conformità ai requisiti del Cyber Resilience Act

Jago Revrenna
Università degli Studi di Trento

Giugno 2025

Indice

1 Introduzione e contesto

Il Regolamento (UE) 2024/2847, noto come **Cyber Resilience Act (CRA)**, sarà ufficialmente adottato dall'Unione Europea a partire dal 2028 per affrontare in modo sistematico le crescenti minacce alla sicurezza informatica, derivanti dalla diffusione di dispositivi connessi e software vulnerabili.
Il CRA introduce un insieme di **requisiti essenziali di cibersicurezza** per tutti i **prodotti con elementi digitali** immessi sul mercato europeo, con l'obiettivo di migliorarne la sicurezza lungo l'intero ciclo di vita, dalla progettazione fino alla dismissione.

1.1 Motivazioni e obiettivi del CRA

Lo scopo è rispondere a una crescente preoccupazione per la sicurezza informatica, data l'esponenziale crescita dei dispositivi connessi e l'aumento degli attacchi ad essi, che comportano conseguenze gravi in termini economici, sociali e di sicurezza per imprese e cittadini.

Il CRA nasce per:

- Rafforzare la **ciberresilienza** dei prodotti digitali;
- Garantire un livello minimo di sicurezza per tutti i dispositivi connessi;
- Migliorare la **trasparenza** lungo la filiera produttiva;
- Facilitare la **conformità normativa** da parte dei produttori.

1.2 Elementi chiave del Regolamento

I punti principali sono:

- **Sicurezza dei prodotti**: I dispositivi devono essere progettati per ridurre i rischi informatici fin dalla fase di sviluppo e garantire che restino sicuri durante tutto il loro ciclo di vita.
- **Gestione delle vulnerabilità**: I produttori sono obbligati a monitorare i loro prodotti per rilevare eventuali vulnerabilità e a fornire aggiornamenti (patch) per correggerle, anche dopo la vendita.
- **Trasparenza**: Sono richieste informazioni dettagliate sui componenti del prodotto (tramite la distinta base del software, SBOM).
- **Certificazione e controllo**: I prodotti devono essere certificati per garantire che soddisfino gli standard di sicurezza. Le autorità competenti sorvegliano il rispetto delle norme, non rilasciando certificazioni in caso di non conformità.

1.3 Struttura del Regolamento

Il Cyber Resilience Act è articolato in due componenti principali:

- Una **parte normativa**, composta da **articoli** suddivisi in capi (titoli), che definisce gli obblighi legali, i soggetti coinvolti, le procedure di conformità, le sanzioni, ecc.

- Una **parte tecnica**, costituita da **allegati**, che elenca i requisiti concreti che i prodotti devono rispettare per essere conformi al Regolamento.

La parte tecnica è composta da **allegati**, ai quali ci si riferirà in questo modo: *Allegato X, Parte X, Punto X* (ad esempio *Allegato I, Parte 1, Punto 2*).

Ad esempio, l'**Allegato I** specifica i **requisiti essenziali di cibersicurezza** ed è suddiviso in due parti:

- **Parte I – Requisiti relativi al prodotto**: include requisiti che riguardano direttamente il design, lo sviluppo e la produzione sicura del dispositivo. I punti coprono aspetti come configurazione sicura, gestione degli accessi, integrità e riservatezza dei dati, resilienza agli attacchi e logging delle attività.
- **Parte II – Requisiti di gestione delle vulnerabilità**: stabilisce obblighi specifici per i fabbricanti relativi al ciclo di vita delle vulnerabilità, come la redazione della distinta base del software (SBOM), la divulgazione responsabile, la fornitura di aggiornamenti di sicurezza e la comunicazione efficace agli utilizzatori.

2 Obiettivi

L'obiettivo è **definire e sviluppare procedure operative per la verifica della conformità** dei dispositivi elettronici ai requisiti previsti dal Regolamento (UE) 2024/2847. Sono stati analizzati gli allegati presenti nel documento e sono stati tecnicizzati i più interessanti.

L'attività prevede:

- La progettazione di un **framework di verifica** a livello procedurale e di processo.
- La definizione di **metodi di test** ripetibili, tracciabili e standardizzati.
- La **validazione** del framework attraverso l'analisi di un caso concreto: la videocamera Wyze v3.

3 Metodo di lavoro

Poichè è necessario avere una struttura solida per affrontare la fase sperimentale, è stata sviluppata una tabella che raccoglie i requisiti tecnici del CRA. In particolare, la sua struttura è la seguente:

- **Allegato / Parte / Punto**: identificativo normativo del requisito.
- **Requisito**: descrizione testuale riassunta direttamente dal CRA.
- **Metodo per implementarlo**: proposta di strategia tecnica o procedurale per la verifica.
- **Nello specifico**: descrizione operativa concreta e strumenti suggeriti.

Per alcuni punti, il campo delle indicazioni specifiche è stato lasciato vuoto: i requisiti in questione sono stati ritenuti troppo complessi o poco rilevanti per la fase attuale.

La tabella rappresenta dunque una prima formalizzazione dei requisiti in chiave operativa, funzionale alla costruzione del framework.

4 Prossimi passi

Nei prossimi step del progetto è prevista l'applicazione di alcuni dei metodi proposti nella tabella a un caso reale: la videocamera Wyze v3. Questa fase permetterà di:

- Verificare l'efficacia e la completezza del framework proposto.
- Valutare la compatibilità con dispositivi consumer (come la videocamera in questione).
- Identificare eventuali criticità o punti di miglioramento.

5 Conclusione

Questo documento rappresenta un passaggio intermedio fondamentale tra la fase teorica e quella pratica del progetto.
L'obiettivo finale è fornire un modello flessibile ed estendibile, che possa supportare la valutazione della compliance normativa e della sicurezza di una varietà di dispositivi elettronici.

6 Allegato - Tabella

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	1	I prodotti devono essere progettati, sviluppati e prodotti in modo da garantire un livello adeguato di cibersicurezza in base ai rischi	• Analizzare (se possibile) il ciclo di sviluppo del prodotto (Secure Software Development Lifecycle) • Verificare la presenza di analisi dei rischi (con tabella rischio = impatto * probabilità) • Controllare se sono previsti/effettuati test di sicurezza di vario genere	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	2a	I prodotti sono messi a disposizione senza vulnerabilità sfruttabili note	• Controllare che le tecnologie utilizzate non presentino vulnerabilità note (effettuare un confronto con un database contenente queste vulnerabilità, il quale viene tenuto aggiornato) • Controllare che la documentazione presenti certificazioni e/o risultati di test effettuati	1. Richiedere al produttore il file SBOM 2. Verificare che elenchi tutti i componenti (librerie, moduli, sistema operativo) con relativa versione e licenza 3. Usare uno strumento di vulnerability scanning che controlla, per ogni componente della SBOM, se la sua versione ha delle criticità (il confronto viene effettuato con il database NVD o un generico database CVE) 4. Riportare l'indice CVSS (Common Vulnerability Scoring System) per ogni componente confrontato, andando ad evidenziare quelli più critici 5. Controllare se sono state rilasciate patch per il componente in questione 6. Controllare quali ambiti coprono i test effettuati dal produttore e di quali certificazioni dispone il prodotto in questione, anche in merito alle eventuali vulnerabilità trovate al punto 3 e 4

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	2b	I prodotti sono messi a disposizione con configurazione sicura in modo predefinito e con la possibilità di effettuare il ripristino	• Analizzare le impostazioni di default (ad esempio: password robuste, porte chiuse, aggiornamenti automatici attivati) • Verificare se c'è la possibilità di effettuare il reset ai dati di fabbrica • Controllare che i servizi non necessari siano disabilitati (di default)	1. Utilizzare uno script o un tool (ad esempio ho trovato Selenium per il test automatico) che consenta di testare la complessità della password, inviando password ed analizzando la risposta 2. Controllare nella documentazione ufficiale se viene menzionata la parola "password", in modo da poter ottenere ulteriori informazioni in merito (lunghezza, caratteri speciali, ...) 3. Eseguire una scansione delle porte TCP/UDP esposte dal prodotto all'avvio, ad esempio con il tool nmap o simili 4. Controllare nella documentazione ufficiale se vengono menzionate parole riguardanti l'argomento porte, in modo da poter ottenere ulteriori informazioni in merito (quali porte sono esposte, quali servizi sono previsti, ...) 5. Intercettare le richieste HTTP/HTTPS che vengono effettuate, usando sniffer di rete come wireshark, con lo scopo di trovare chiamate a server di aggiornamento automatico (quindi qualcosa in stile polling). 6. Nuovamente, controllare la documentazione ufficiale e verificare se viene menzionato l'argomento "aggiornamenti automatici", in modo da poter ottenere ulteriori informazioni in merito (frequenza del controllo aggiornamenti, ...) 7. Controllare la documentazione ufficiale e verificare se viene menzionato l'argomento "reset ai dati di fabbrica", in modo da poter ottenere ulteriori informazioni in merito (come effettuarlo, ...) 8. Per quanto riguarda la verifica dei servizi non necessari disabilitati, oltre a controllare la documentazione come nei punti precedenti, si potrebbe (nel caso in cui il dispositivo sia online / disponga di una cli) vedere cosa è in ascolto sulla rete (usando ad esempio nmap/netstat come prima, oppure ps/top se è possibile usare la cli). In base ai risultati, classificarli come necessari o non necessari.

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	2c	Garantire la gestione delle vulnerabilità tramite aggiornamenti di sicurezza (preferibilmente automatici) con possibilità di disattivarli e notificarli	• Verificare se è prevista la possibilità di effettuare gli aggiornamenti in modo automatico • Controllare se è possibile disattivare gli aggiornamenti automatici (se si, con che facilità) • Verificare in che modo gli aggiornamenti vengono notificati all'utente (notifiche push, notifiche che impediscono l'utilizzo del dispositivo, ...)	1. Intercettare le richieste HTTP/HTTPS effettuate automaticamente dal dispositivo, usando uno sniffer di rete (es. Wireshark, tcpdump,...) per verificare se il dispositivo contatta server di aggiornamento subito dopo l'avvio (come descritto prima). 2. Controllare nella documentazione ufficiale se viene menzionato l'argomento "aggiornamenti automatici", così da poter capire se gli aggiornamenti automatici sono abilitati di default e con quale frequenza viene effettuato il controllo. 3. Utilizzare strumenti come Selenium, curl, ... per poter effettuare test mediante (ad esempio) uno script che accede al pannello di configurazione, per poi cercare un'opzione/flag per disattivare gli aggiornamenti automatici. 4. Valutarne la complessità. In alternativa, analizzare la documentazione ufficiale e cercare termini come "disattiva aggiornamenti, per capire se e come è prevista la disattivazione da parte dell'utente. 5. Cercare nella documentazione ufficiale se e come viene notificato un aggiornamento all'utente, ad esempio tramite messaggi sullo schermo, popup, notifiche push, oppure se il dispositivo impedisce l'uso fino all'aggiornamento.

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	2d	Garantire la protezione dall'accesso non autorizzato tramite adeguati controlli di login e in che modo viene verificata l'identità dell'utente	• Verificare in che modo viene implementata l'autenticazione (es. password che devono essere complesse, MFA, ...) • Controllare sistemi di gestione degli accessi e logging accessi non autorizzati. • Analizzare policy di gestione account e privilegi.p.	1. Verificare in che modo viene implementata l'autenticazione: se è disponibile un'interfaccia web o CLI, utilizzare uno script (ad esempio Selenium o expect, come descritto prima) che invia tentativi di login con credenziali errate o deboli. Osservare quindi se il sistema impone vincoli di sicurezza (es. lunghezza minima della password, caratteri speciali, ...). Analizzare la documentazione ufficiale cercando frasi come "password complexity" o "MFA"/"autenticazione a due fattori", per capire se sono previsti meccanismi di autenticazione avanzati. 2. Analizzare la documentazione ufficiale per individuare la presenza di un sistema di gestione degli accessi. Verificare se sono previsti account con permessi differenti, come ad esempio utenti normali e amministratori. Controllare inoltre se viene registrata l'attività di login, cercando parole chiave come "access/security logs" o "login audit". Se questi log sono accessibili, si può simulare un tentativo di accesso errato per vedere se l'evento viene registrato nei log o segnalato in qualche modo. 3. Simulare un attacco brute-force per verificare se il sistema applica contromisure, eseguendo tentativi di login errati con lo stesso account tramite strumenti di testing descritti sopra. Osservare se il sistema reagisce bloccando temporaneamente l'account, introducendo un ritardo crescente tra i tentativi o generando un log di qualche tipo. 4. Verificare se vengono specificate regole come il numero massimo di tentativi di login falliti prima del blocco, il timeout automatico della sessione dopo inattività e l'obbligo di cambiare le credenziali periodicamente. Si potrebbe confrontare i valori dichiarati con quelli raccomandati da standard noti, ad esempio NIST. 5. Consultare la documentazione o testare direttamente il comportamento del sistema per capire se esistono ruoli distinti e se l'accesso a funzionalità critiche è riservato solo agli utenti autorizzati. Se disponibile un'interfaccia, provare ad accedere con un utente non privilegiato e verificare se le funzioni amministrative sono nascoste o inaccessibili.

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	2e	Prevedere riservatezza dei dati conservati/inviati/trattati tramite la crittografia (o altri mezzi tecnici)	• Verificare la presenza di crittografia dei dati conservati e in transito (ad esempio criptandoli con AES e inviandoli con TLS) • Controllare se vengono utilizzati protocolli sicuri per le comunicazioni • Controllare come avviene la gestione delle chiavi di cifratura.	1. Catturare il traffico di rete generato dal dispositivo durante il funzionamento utilizzando strumenti come Wireshark o tcpdump. 2. Analizzare i pacchetti per verificare che i dati sensibili non siano trasmessi in chiaro, identificando i protocolli usati e controllando che non siano non cifrati (es HTTP invece di HTTPS, ...). Successivamente esaminare i certificati TLS per versione e validità della cifratura. 3. Eseguire una scansione automatizzata verso le porte di rete del dispositivo usando strumenti come nmap con lo script ssl-enum-ciphers per raccogliere informazioni sulle versioni TLS supportate, le cipher suite disponibili e la configurazione di sicurezza, confrontando i risultati con best practice di sicurezza. 4. Estrarre informazioni sulla crittografia tramite API, interfacce web o endpoint accessibili, interrogando le risorse disponibili per rilevare configurazioni o report che indicano se i dati sono protetti da cifratura e quali algoritmi sono utilizzati. Analizzare le risposte (ma anche la documentazione) cercando parole chiave come “encryption”, “AES”, ... e valutare la presenza e il tipo di cifratura. 5. Interrogare le API o interfacce di gestione chiavi esposte dal dispositivo per verificare come avviene la conservazione e la protezione delle chiavi di cifratura, controllando se sono previste politiche di rotazione, backup sicuro e restrizioni di accesso. 6. Integrare un proxy Man-in-the-Middle (ad esempio mitmproxy o Burp Suite) che intercetti il traffico cifrato: questo verificherà che la connessione TLS fallisca in presenza di corretta validazione certificati.

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	2f	Prevedere integrità dei dati, dei programmi e delle configurazioni da manipolazioni e modifiche non autorizzate, segnalando le corruzioni	• Verificare l'uso della firma digitale per i dati sensibili. • Utilizzare il log per tracciare tutte le modifiche che vengono effettuate (e da chi) • Utilizzare funzioni di hashing per effettuare check sull'integrità dei dati (specialmente quelli trasportati)	
1	1	2g	Trattare solo dati adeguati, pertinenti e limitati a quanto necessario (minimizzazione dei dati).	• Verificare che i dati raccolti siano limitati alla funzionalità del prodotto (principio del privilegio minimo), cosi anche come i privilegi che vengono assegnati alle varie parti del software • E' presente documentazione della gestione dati?	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	2h	Proteggere la disponibilità delle funzioni essenziali dopo un incidente, anche contro attacchi DoS	• Verificare se sono presenti accorgimenti/sistemi per limitare attacchi DoS/DDoS, come i Captcha oppure metodi di rate limiting • Verificare se ci sono piani di resilienza (come server ridondanti) oppure piani di disaster recovery (per ripristinare la situazione iniziale dopo eventuali attacchi/emergenze)	1. Interrogare le API o l'interfaccia web per rilevare la presenza di sistemi anti DoS/DDoS, verificando nei metadati delle risorse critiche (login, API di scrittura, form) l'attivazione di CAPTCHA o flag di rate limiting. 2. Inviare un numero elevato di richieste sequenziali agli endpoint critici e analizzare le risposte HTTP. In caso di risposta Too Many Requests o errori di blocco temporaneo, è confermata l'efficacia del rate limiting. 3. Interrogare le API di backup, se presenti, per ottenere informazioni in merito a backup e recovery plan.
1	1	2i	Ridurre al minimo l'impatto (negativo) che potrebbe avere il prodotto in questione sulla disponibilità dei servizi di altri dispositivi o reti	• Verificare in che modo il prodotto consuma risorse, sia in condizioni normali che in caso di errori • Controllare che il dispositivo non generi traffico di rete anomalo	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	2j	Limitare le superfici di attacco, comprese le interfacce esterne.	• Verificare che siano esposte solamente le interfacce necessarie (ovviamente con le dovute precauzioni) • Controllare che porte e protocolli non necessari siano disabilitati (principio del privilegio minimo anche qui), cosi come assicurarsi che la configurazione delle API eventualmente utilizzate avvenga in modo sicuro	
1	1	2k	Ridurre l'impatto degli incidenti utilizzando meccanismi di attenuazione dello sfruttamento.	• Verificare la presenza di protezioni come ASLR, DEP, stack canaries (non conosco queste tecnologie, mi sono informato da internet) • Controllare che ci siano meccanismi di sandboxing e rate limiting, così come controllare che gli errori vengano gestiti in modo corretto	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	1	2l	Fornire informazioni sulla sicurezza registrando/monitorando le attività effettuate, con possibilità di disattivazione da parte dell'utente	• Verificare se è presente un sistema di log delle operazioni sensibili (o delle operazioni in generale) e che questo sistema sia sicuro (dal punto di vista crittografico) • Controllare che l'utente possa attivare/disattivare il sistema di log descritto sopra	
1	1	2m	Offrire possibilità all'utente di rimuovere permanentemente dati e impostazioni in modo sicuro e che possa trasferirli in modo sicuro	• Verificare se sono implementate funzioni di wipe sicuro dei dati e se l'utente può usufruirne • Controllare se esiste una funzione di esportazione sicura dei dati (in che modo li esporta, verso dove, . . .) • Controllare che i dati cancellati non siano recuperabili facilmente	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	2	1	I fabbricanti identificano e documentano vulnerabilità e componenti, riassumendole in una distinta base software. Questa deve includere le dipendenze di primo livello del prodotto	• Verificare la presenza di una Software Bill of Materials (SBOM) aggiornata, il cui formato deve essere leggibile da dispositivi automatici • Controllare se esiste un sistema per tracciare le vulnerabilità interne descritte nella SBOM	1. Interrogare l'API/endpoint dedicato per scaricare la SBOM in formato leggibile da dispositivi automatici (ad es. XML/JSON, ...): la SBOM deve essere conforme agli standard, disponendo dei campi obbligatori e dello schema JSON/XML (cosa da controllare). 2. Estrarre dalla SBOM la lista dei componenti e delle dipendenze (quantomeno di primo livello), controllando che ogni voce includa nome, versione e identificatori univoci. 3. Interrogare le API del sistema di gestione vulnerabilità (se presenti) per ciascun componente elencato nella SBOM, raccogliendo elenco di CVE e relativi punteggi.

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	2	2	I fabbricanti correggono tempestivamente le vulnerabilità e rilasciano aggiornamenti di sicurezza separati dalla funzionalità.	• Verificare la politica di gestione degli aggiornamenti di sicurezza, controllando anche la frequenza di rilascio delle patch: per quanto tempo, ogni quanto tempo, ... • Controllare che gli aggiornamenti di sicurezza sono separati dagli aggiornamenti funzionali	1. Interrogare l'API o consultare la documentazione ufficiale per recuperare la policy di gestione degli aggiornamenti, individuando il periodo di supporto e la frequenza delle patch. 2. Estrarre dalla policy i valori specifici di inizio supporto, fine supporto, intervallo massimo tra release di sicurezza e tempo obiettivo per il rilascio di patch critiche, confrontandoli con i requisiti normativi (per esempio, entro 30 giorni dalla scoperta della vulnerabilità). 3. Interrogare l'API dei rilasci o sfogliare il changelog ufficiale per ottenere la lista cronologica di tutte le release. Parsare per ciascuna release i metadati di rilascio, tipo e identificativo univoco. Verificare se è specificato il tipo di update (ad esempio feature o security). 4. Effettuare una richiesta di download di un security update o scaricare il pacchetto dalla documentazione, quindi ispezionare il manifest interno / note di rilascio per accertare che il changelog contenga solo correzioni di sicurezza e non nuove funzionalità.
1	2	3	Vengono effettuate prove e riesami efficaci e periodici della sicurezza del prodotto	• Verificare se sono descritte le tipologie di test effettuati e la loro frequenza • Controllare report con risultati dei test • Accertarsi che esista un calendario ufficiale di riesami	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	2	4	Pubblicare e divulgare pubblicamente le informazioni sulle vulnerabilità risolte	• Esaminare le Release Notes pubblicate sul sito del produttore: devono contenere descrizione vulnerabilità, prodotti interessati, gravità e istruzioni di mitigazione • Controllare date di pubblicazione in relazione alla disponibilità della patch	
1	2	5	Divulgazione in modo coordinato le vulnerabilità	• Verificare che sia descritto il modo in cui si intende gestire la divulgazione coordinata delle vulnerabilità (ad esempio tramite database delle vulnerabilità, sito/forum ufficiale, apposita pagina, ...)	
1	2	6	Devono esserci misure prese per facilitare la segnalazione di vulnerabilità e contatto dedicato	• Cercare nel manuale o sul sito l'indirizzo e-mail o link dedicato per segnalazioni • Verificare la presenza di istruzioni chiare e tempi di risposta stimati	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
1	2	7	Meccanismi per distribuire in modo sicuro gli aggiornamenti (inclusi automatici)	• Controllare la documentazione in merito alla sicurezza nella distribuzione degli aggiornamenti e modalità di fallback • Verificare che siano descritti i criteri di roll-out e i requisiti di autenticazione per ottenere l'aggiornamento • Accertarsi della presenza di istruzioni per l'attivazione/disattivazione del meccanismo automatico	
1	2	8	Garanzia di disponibilità tempestiva e gratuita degli aggiornamenti di sicurezza (con messaggi di avviso agli utilizzatori)	• Controllare i "Security Bulletins" per i tempi di rilascio (ad es. entro X giorni dalla scoperta) • Verificare esempi di comunicazioni (e-mail, notifiche) in merito alla distribuzione degli update	
7	/	1a	Descrizione generale del prodotto con elementi digitali: finalità prevista	• Controllare la descrizione delle finalità previste per il prodotto	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
7	/	1b	Descrizione delle versioni software rilevanti per la conformità ai requisiti essenziali di cibersicurezza	• Assicurarsi che sia mantenuta una SBOM aggiornata con tutte le librerie (e relative versioni) • Controllare la presenza di un registro delle release con indicazione delle patch di sicurezza che ogni release include/prevede	
7	/	1c	Documentazione hardware: fotografie, illustrazioni e schemi delle caratteristiche esterne e interne del prodotto	• Controllare se sono presenti foto/illustrazioni/schemi delle viste esterne e interne • Prestare particolare attenzione a porte, antenne e numero seriale	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
7	/	1d	Informazioni e istruzioni dettagliate per l'utilizzatore, inclusi installazione, configurazione e aggiornamenti (basandosi su allegato II)	Controllare la presenza di: • Dati fabbricante (nome/marchio, contatti, sito) • Punto unico segnalazioni e modo in cui il fabbricante divulga in modo coordinato le vulnerabilità • Identificativo univoco (modello e numero seriale) • Finalità, funzionalità chiave e rischi d'uso • Tipo di supporto sicurezza + data fine assistenza • Istruzioni per: avvio sicuro, installazione aggiornamenti, smantellamento sicuro (eliminazione sicura dei dati degli utilizzatori), disattivazione degli aggiornamenti automatici e integrazione con altri sistemi	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
7	/	2a	Documentazione di progettazione, sviluppo e architettura del sistema: componenti, flussi dati e dipendenze software	• Controllare se l'architettura viene descritta mediante diagrammi di componenti (component diagram) e sequence (flussi dei dati, diagrammi BPMN?, API utilizzate e routes/endpoint esposti, dipendenze sotfware) • Assicurarsi che punti di integrazione (protocolli usati per la comunicazione, interfacce, formati dei dati, meccanismi di autenticazione) e librerie di terze parti utilizzate siano elencati e descritti	
7	/	2b	Gestione delle vulnerabilità: SBOM, policy di disclosure e canale dedicato per la segnalazione e aggiornamenti	Assicurarsi che sia: • Descritta la SBOM nella documentazione • Definito il processo di rilascio degli aggiornamenti (ad esempio OTA firmati e distribuiti via HTTPS, ...) • Presente un link/contatto per la segnalazione di vulnerabilità • Descritto il modo in cui si intende gestire la divulgazione coordinata delle vulnerabilità (ad esempio tramite database delle vulnerabilità, sito/forum ufficiale, apposita pagina, ...)	

All.	Parte	Punto	Requisito	Metodo per implementarlo	Nello specifico
7	/	2c	Processi di produzione, monitoraggio e convalida dei controlli qualità	Verificare che siano presenti: • Sezioni che elencano le fasi di assemblaggio e collaudo (dall'inizio alla fine) • Elenco degli strumenti/metodologie impiegate • Descrizione delle procedure di gestione dei log • Date e versioni del software/hardware testato	
7	/	3	Valutazione completa dei rischi di cibersicurezza e definizione di un piano di mitigazione	• Verificare che sia stato effettuato un Risk Assessment con matrice impatto x probabilità • Controllare che siano stati allineati i controlli richiesti dall'Allegato I parte I ai rischi identificati • Controllare che sia stato redatto un piano di mitigazione	