

IL DIRETTORE GENERALE ALL'ORGANIZZAZIONE, PERSONALE, SISTEMI
INFORMATIVI E TELEMATICA

Visto il Decreto Legislativo 30 giugno 2003, n. 196 "Codice in materia di protezione di dati personali";

Viste le deliberazioni della Giunta regionale:

- n. 960 del 27/06/2005 con cui è stata adottata la Direttiva in materia di trattamento di dati personali;
- n. 1264 del 01/08/2005 con cui sono state adottate le Linee guida della Giunta della Regione Emilia-Romagna in materia di protezione dei dati personali;

Vista inoltre la propria determinazione n. 1033/2006 "Disciplinare tecnico in materia di sicurezza delle applicazioni informatiche nella Giunta della Regione Emilia-Romagna" che definiva il carattere sperimentale delle prescrizioni in esso contenute.

Considerato che la sperimentazione, condotta per la durata di un anno, non ha evidenziato criticità tali da prevedere una modifica delle prescrizioni già individuate.

Sentito il parere del Comitato di Direzione nella seduta del 26/02/2007;

Dato atto di aver rispettato le vigenti disposizioni in materia di relazioni sindacali;

Dato atto del parere di regolarità amministrativa espresso dal Responsabile della Sicurezza della Giunta, ai sensi della deliberazione della Giunta Regionale n. 960/2005;

DETERMINA

1. di approvare l'allegato "Disciplinare tecnico in materia di sicurezza delle applicazioni informatiche nella Giunta della Regione Emilia-Romagna" che conferma sostanzialmente le prescrizioni precedentemente individuate;
2. di applicare all'interno delle proprie strutture l'allegato "Disciplinare tecnico in materia di sicurezza delle applicazioni informatiche nella Giunta della Regione Emilia-Romagna";
3. di procedere alla diffusione del contenuto dell'allegato "Disciplinare tecnico in materia di sicurezza delle applicazioni informatiche nella Giunta della Regione Emilia-Romagna" a partire dalla data di approvazione del presente atto anche ai Responsabili esterni fornitori di servizi di sviluppo di applicazioni informatiche.

IL DIRETTORE GENERALE
(Gaudenzio Garavini)

Allegato

**Disciplinare tecnico in materia di sicurezza
delle applicazioni informatiche nella
Giunta della Regione Emilia-Romagna**

INDICE

1. Premessa	3
2. Applicabilità	3
3. Principi generali	3
3.1 Applicazioni sicure	3
3.2 Architettura applicativa	3
4. Design e sviluppo dell'applicazione	4
4.1 Analisi dei requisiti e design	4
4.2 Autenticazione	5
4.3 Autorizzazione	7
4.4 Validazione dei dati	8
4.5 Gestione delle sessioni utente	8
4.6 Logging	9
4.7 Crittografia e disponibilità dei dati	10
5. Test, deployment e gestione dell'applicazione	11
6. Requisiti minimi previsti dalla normativa vigente	11
Appendice A - Esempio di analisi dei rischi per applicazioni web	13
1. Stabilire l'obiettivo di sicurezza	13
2. Descrivere sinteticamente l'applicazione	13
3. Minacce	13
4. Meccanismi di sicurezza da implementare	13
Appendice B: Liste di controllo	15
B.1 Design e sviluppo dell'applicazione	15
B.2 Test, deployment e gestione dell'applicazione	19
B.3 Requisiti minimi previsti dalla normativa vigente	20
Appendice C: Glossario	21

1. Premessa

Il presente disciplinare descrive gli aspetti tecnici e procedurali richiesti per il design, lo sviluppo, il deployment, il test e la gestione di un'applicazione sicura. Particolare riguardo è dedicato alle **applicazioni web**, in quanto maggiormente esposte a minacce per la loro caratteristica intrinseca di rendere disponibili servizi ad un numero elevato, e spesso indefinito, di utenti.

2. Applicabilità

Il disciplinare si applica all'interno della Giunta della Regione Emilia-Romagna (di seguito denominata Ente) quale strumento di riferimento per i soggetti incaricati di:

- progettare un'applicazione;
- sviluppare un'applicazione;
- acquistare un'applicazione;
- valutare/scegliere fornitori di servizi di sviluppo applicazioni;
- testare la sicurezza di applicazioni;
- adeguare un'applicazione ai criteri di sicurezza previsti dalla normativa vigente;
- installare, gestire o mantenere un'applicazione.

3. Principi generali

3.1 Applicazioni sicure

I destinatari del presente disciplinare devono considerare le minacce di sicurezza e le contromisure disponibili relativamente a dati e informazioni, secondo le indicazioni fornite nei paragrafi seguenti. Tali indicazioni si basano sul fondamento che un'applicazione è sicura quando è in grado di preservare *confidenzialità*, *integrità* e *disponibilità* delle risorse, assicurando costantemente:

- l'identificazione dell'utente che accede alle risorse;
- la limitazione degli accessi alle risorse;
- la comunicazione sicura con l'esterno;
- la conservazione sicura dei dati.

3.2 Architettura applicativa

L'architettura di riferimento di un'applicazione si compone logicamente di tre livelli distinti:

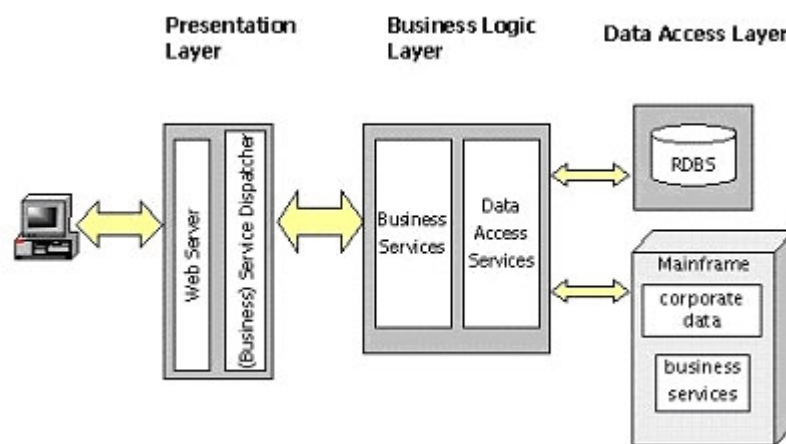
- **livello di presentazione o interfaccia utente**, per la rappresentazione dei dati verso l'utente e della raccolta e verifica dei dati in ingresso;
- **livello business logic o applicazione**, per l'implementazione della logica di elaborazione dei dati. Acquisisce i dati dal livello presentazione e dal livello data

access, esegue elaborazioni su di essi e li restituisce elaborati ai livelli di presentazione e data access;

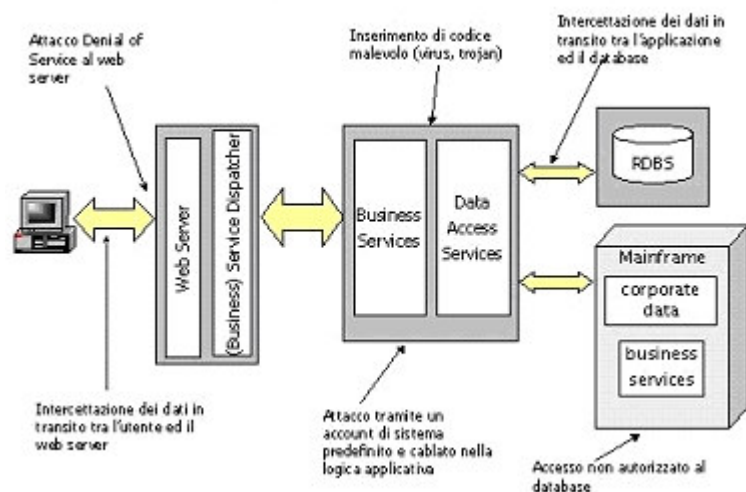
- **livello data access**, per l'accesso alle basi dati, per esempio basi dati permanenti come database relazionali, ma anche servizi di accesso a dati dinamici.

Gli attuali tool di sviluppo ed i componenti middleware sono generalmente modellati secondo questo schema architetturale. Le applicazioni devono, quantomeno a livello logico, seguire questo approccio di separazione dei livelli per garantire compartimentazione, separazione di privilegi, e modularità del software.

L'architettura logica di una applicazione distribuita può essere quindi schematizzata come segue:



A titolo di esempio, vengono illustrate alcune minacce tipiche dei tre livelli:



4. Design e sviluppo dell'applicazione

4.1 Analisi dei requisiti e design

Minacce

Accesso non autorizzato alle risorse, esecuzione di operazioni non consentite, mancata disponibilità dei servizi e dei dati, mancato rispetto degli obblighi previsti dalla [normativa vigente](#).

Contromisure

- A. Identificare, in fase di raccolta dei requisiti, la natura ed il valore dei dati e delle informazioni che saranno trattate dall'applicazione. Individuare in particolare, facendo riferimento alla normativa vigente, se si trattino le seguenti categorie di [dati personali](#):
- dati personali non sensibili e giudiziari;
 - dati [sensibili](#);
 - dati [giudiziari](#).
- B. Eseguire, successivamente all'individuazione della natura e del valore dei dati e delle informazioni, l'analisi dei rischi incombenti su di esse in relazione alle minacce ed alle contromisure di sicurezza disponibili. L'analisi dei rischi è elemento fondamentale per la scelta delle misure di sicurezza appropriate. Tale scelta deve essere fatta in funzione del valore delle risorse da proteggere e dei danni che una eventuale compromissione della sicurezza comporterebbe. L'[Appendice A](#) contiene un esempio sintetico di analisi dei rischi effettuata per un'applicazione web.
- C. Considerare eventuali vincoli infrastrutturali e tecnologici che possono influire sul comportamento dell'applicazione e comprometterne la sicurezza (per es. restrizione di porte o protocolli, piattaforme tecnologiche di sviluppo, ecc.).
- D. Censire con precisione, in fase di design, le porte ed i protocolli utilizzati dall'applicazione, in modo da predisporre un ambiente di produzione che limiti all'indispensabile la [superficie di attacco](#) dell'applicazione stessa.
- E. Progettare l'applicazione prevedendo l'implementazione dei seguenti [meccanismi di sicurezza](#):
- autenticazione;
 - autorizzazione;
 - validazione dei dati;
 - gestione delle sessioni utente;
 - logging;
 - crittografia e protezione dei dati.
- F. Utilizzare la lista di controllo "Analisi dei requisiti e design" ([Appendice B.1](#)) come strumento di supporto per verificare la rispondenza dell'applicazione a quanto richiesto nei punti precedenti.

4.2 Autenticazione

Minacce

Accesso non autorizzato alle risorse, [spoofing](#), individuazione/furto delle password, [privilege escalation](#).

Contromisure

- A. Stabilire in fase di design dove ed in che modo sia necessario garantire l'autenticazione (di utenti o entità). Tale scelta deve essere fatta in funzione del valore delle risorse da proteggere e delle policy dell'Ente in materia di autenticazione.
- B. Stabilire il meccanismo di autenticazione e la scelta delle credenziali da utilizzare, in funzione del valore delle risorse da proteggere. Le credenziali di autenticazione possono consistere in un codice per l'identificazione dell'utente associato a una parola chiave riservata conosciuta solamente dal medesimo (per es. userid/password) oppure in un dispositivo di autenticazione in possesso e uso esclusivo dell'utente (per es. smartcard o token hardware), eventualmente associato a un codice identificativo o a una parola chiave, oppure in una caratteristica biometrica dell'utente (per es. impronta digitale), eventualmente associata a un codice identificativo o a una parola chiave.
- C. Utilizzare, dove possibile, i meccanismi di autenticazione centralizzati in uso nell'Ente, in modo che l'autenticazione non sia parte del codice applicativo, ma sia basata su meccanismi dedicati (per es. LDAP, Active Directory, ecc.). In questo modo si realizza una duplice ottimizzazione: da un lato l'utente non è costretto a ricordare una nuova userid/password (all'aumentare del numero di password utilizzate aumenta infatti la probabilità che esse siano dimenticate, annotate per iscritto, scelte con meno cura, utilizzate su più sistemi diversi), dall'altro non si utilizza un database dedicato per la memorizzazione e gestione credenziali a livello applicativo.
- D. Prevedere, in caso di autenticazione basata su password, i seguenti meccanismi di sicurezza obbligatori per legge:
- scadenza della password: prevedere meccanismi di controllo della scadenza della validità password, che deve essere inferiore ai 180 giorni (90 giorni nel caso di applicazioni relative al trattamento di dati sensibili e/o giudiziari);
 - lunghezza della password: prevedere meccanismi di controllo sulla lunghezza della password, che deve essere di almeno 8 caratteri;
 - modifica al primo login: prevedere meccanismi che consentano la modifica della propria password da parte dell'utente al primo accesso al sistema.
- E. Prevedere ulteriori meccanismi di sicurezza per i sistemi basati su password dove valutato necessario dall'analisi dei rischi effettuata. In particolare:
- meccanismi di lockout: prevedere la disabilitazione di un account dopo un intervallo finito di tentativi di accesso non riusciti (per contrastare gli attacchi alle password di tipo **bruteforce**). Prevedere meccanismi di difesa da attacchi di tipo **denial-of-service** causati dal blocco volontario di account legittimi (per es. bloccando un account per poi riabilitarlo trascorsi 10' dal blocco, oppure prevedendo un delay di 5" a seguito di un'autenticazione errata);
 - meccanismi di reset della password: prevedere meccanismi che consentano all'utente di modificare la propria password senza l'intervento degli amministratori di sistema;
 - meccanismi di cifratura delle password: prevedere meccanismi di conservazione protetta delle password, che non devono mai essere conservate o trasmesse in

chiaro (per es. utilizzo di [hash](#) per la memorizzazione delle password, utilizzo di protocolli di comunicazione cifrati come HTTPS, ecc.);

- d. meccanismi di controllo della robustezza delle password: prevedere meccanismi di controllo che consentano esclusivamente l'utilizzo di password corrispondenti a determinati criteri di complessità (per es. password con almeno un valore letterale maiuscolo, con almeno un valore numerico, con almeno un carattere simbolico, ecc.).
- F. Prevedere meccanismi di disattivazione delle credenziali non utilizzate da almeno 180 giorni.
- G. Ridurre al minimo le informazioni fornite in caso di errore di autenticazione, compatibilmente con la necessità di fornire le informazioni necessarie all'utente per la comprensione dell'errore verificatosi (per esempio disabilitando il debugging remoto). Informazioni superflue sono utili ad un attaccante per comprendere i meccanismi di autenticazione del sistema ed i metodi per aggirarli.
- H. Non riutilizzare i codici di identificazione già impiegati assegnandoli ad altri utenti (neanche in tempi diversi).
- I. Utilizzare la lista di controllo "Autenticazione" ([Appendice B.1](#)) come strumento di supporto per verificare la rispondenza dell'applicazione a quanto richiesto nei punti precedenti.

4.3 Autorizzazione

Minacce

Accesso non autorizzato ai dati, modifica non consentita di dati, esecuzione di operazioni non consentite.

Contromisure

- A. Implementare meccanismi di separazione dei privilegi per garantire l'utilizzo delle risorse in funzione di differenti profilazioni degli utenti.
- B. Utilizzare il principio del "minimo privilegio" nell'attribuzione dei permessi, ovvero abilitare l'accesso alle sole risorse indispensabili e negarlo a tutte le restanti.
- C. Limitare al minimo, ed evitare dove possibile, l'accesso alle risorse di sistema (file, cartelle, registry, log, ecc.).
- D. Non consentire al login applicativo l'accesso diretto in scrittura alle tabelle dei database di backend. Utilizzare credenziali di autenticazione ai database con i privilegi minimi indispensabili.
- E. Stabilire, in caso di applicazioni web, dove sia necessario applicare la separazione dei privilegi tra aree web ad accesso pubblico ed aree web ad accesso riservato.
- F. Utilizzare la lista di controllo "Autorizzazione" ([Appendice B.1](#)) come strumento di supporto per verificare la rispondenza dell'applicazione a quanto richiesto nei punti precedenti.

4.4 Validazione dei dati

Minacce

Stringhe “nocive” inserite in query, form, cookie e header HTTP. Esecuzione di comandi, [cross-site scripting](#) (XSS), [SQL injection](#), [buffer overflow](#), denial-of-service.

Contromisure

- A. Validare sia l’input sia l’output per controllare che siano rispondenti a quanto l’applicazione si aspetta in termini di:
- formato dei dati;
 - sintassi;
 - dimensioni.
- B. Considerare sempre inattendibili, e quindi da validare, tutti i dati in input e output.
- C. Effettuare tutte le validazioni dei dati lato server. Dove per comodità vengano implementate validazione lato client, esse devono essere accessorie a quelle effettuate lato server. Le strategie di validazione possibili sono:
- accettare solo dati riconosciuti validi;
 - rigettare i dati riconosciuti come invalidi;
 - modificare i dati invalidi per renderli validi.
- Preferire, dove tecnicamente realizzabile, la soluzione (a) (i dati riconosciuti validi rimangono costanti nel tempo, mentre i dati invalidi possono cambiare nel tempo con l’evolversi delle tecniche di attacco). Impiegare la soluzione (c) solo come misura aggiuntiva ad una delle precedenti. La soluzione ideale comprende tutte e tre le strategie di validazione.
- D. Adottare un sistema di firma digitale per certificare i dati che provengono o vengono inviati ad altre applicazioni ed accettare dati esclusivamente da sistemi riconosciuti e fidati.
- E. Utilizzare la lista di controllo "Validazione dei dati" ([Appendice B.1](#)) come strumento di supporto per verificare la rispondenza dell'applicazione a quanto richiesto nei punti precedenti.

4.5 Gestione delle sessioni utente

Minacce

Spoofing, [session hijacking](#).

Contromisure

- A. Prevedere meccanismi di protezione delle credenziali utilizzate per il riconoscimento dell’utente dopo il login.
- B. Limitare la durata delle sessioni ad un periodo di tempo definito in funzione delle caratteristiche funzionali dell'applicazione e prevedere il blocco della sessione allo scadere di tale periodo.

- C. Non trasferire mai in chiaro gli identificatori di sessione (per es. token, stringhe di query, ecc.). Nelle applicazioni web proteggere i cookie di autenticazione di sessione tramite l'utilizzo del protocollo TLS o cifrandone il contenuto.
- D. Implementare meccanismi di logout che permettano all'utente di forzare la chiusura di una sessione.
- E. Utilizzare la lista di controllo "Gestione delle sessioni utente" ([Appendice B.1](#)) come strumento di supporto per verificare la rispondenza dell'applicazione a quanto richiesto nei punti precedenti.

4.6 Logging

Minacce

Mancato rilevamento di intrusioni, impossibilità di dimostrare un'eventuale azione illecita compiuta da un utente, difficoltà nel diagnosticare malfunzionamenti e anomalie di funzionamento dell'applicazione.

Contromisure

- A. Definire in fase di design quali sono gli eventi chiave per la sicurezza dell'applicazione da rilevare tramite logging.
- B. Registrare nei log, dove tecnicamente possibile, i seguenti eventi applicativi:
 - a. autenticazione applicativa (login e logout, riusciti e non);
 - b. accesso ai dati (lettura e scrittura);
 - c. modifica di funzioni amministrative (per es. la disabilitazione delle funzioni di logging, la gestione dei permessi, ecc.).
- C. Prevedere meccanismi di controllo e modifica del livello di granularità dei dati rilevabili.
- D. Prevedere la possibilità di registrare, all'interno di un voce di log, le seguenti informazioni:
 - a. data/ora dell'evento;
 - b. luogo dell'evento (per es. macchina, indirizzo IP, ecc.);
 - c. identificativo dell'entità che ha generato l'evento (per es. utente, servizio, processo, ecc.);
 - d. descrizione dell'evento.
- E. Prevedere meccanismi di conservazione dei log in file su cui sia possibile effettuare esclusivamente la scrittura incrementale o su supporti non riscrivibili (per es. CD-R). Conservare i log, dove tecnicamente possibile, su sistemi dedicati.
- F. Prevedere meccanismi di backup dei log secondo le procedure di backup centralizzato previste dall'Ente.
- G. Prevedere meccanismi di sovrascrittura dei log esistenti ad intervalli regolari. La frequenza di rotazione dei log è dipendente dalla frequenza di generazione degli eventi dell'applicazione. Devono in ogni caso essere rispettate le policy dell'Ente in materia di trattamento dei log.

- H. Prevedere meccanismi di controllo degli accessi ai log tramite autenticazione ed autorizzazione. L'accesso ai log deve poter essere eseguito solo da utenti privilegiati (per es. membri del gruppo *Administrators* nei sistemi Windows based o *root* nei sistemi UNIX) e comunque, salvo esigenze particolari e documentate, secondo le policy dell'Ente in materia di trattamento dei log.
- I. Prevedere meccanismi di verifica periodica del corretto funzionamento dei sistemi di logging.
- J. Utilizzare la lista di controllo "Logging" ([Appendice B.1](#)) come strumento di supporto per verificare la rispondenza dell'applicazione a quanto richiesto nei punti precedenti.

4.7 Crittografia e disponibilità dei dati

Minacce

Accesso non autorizzato ai dati, accesso a credenziali utenti, mancata disponibilità dei dati.

Contromisure

- A. Considerare l'utilizzo di meccanismi di cifratura dei dati (sia per la conservazione sia per la trasmissione) in funzione del valore delle risorse da proteggere e delle minacce di sicurezza, con particolare attenzione ai casi in cui l'applicazione sia destinata al trattamento di dati sensibili e/o giudiziari.
- B. Utilizzare meccanismi di firma digitale nei casi in cui sia necessario garantire la non ripudiabilità delle transazioni. Prevedere, in tal caso, chiavi di firma distinte dalle chiavi di cifratura.
- C. Non conservare né trasmettere in chiaro password e credenziali di autenticazione.
- D. Definire, in fase di design dell'applicazione, gli algoritmi da utilizzare e la lunghezza delle chiavi di cifratura in funzione dell'importanza delle risorse da proteggere e del progresso tecnico nel campo della sicurezza informatica e della crittoanalisi.
- E. Utilizzare algoritmi di cifratura standard e chiavi di lunghezza adeguata. A titolo di esempio:
 - a. algoritmi di cifratura simmetrici: TripleDES (3DES) o IDEA con chiavi a 256 bit o oltre, Advanced Encryption Standard (AES) con chiavi a 128 bit o oltre;
 - b. algoritmi di cifratura asimmetrici: Rivest-Shamir-Adleman (RSA), ElGamal o Digital Signature Algorithm (DSA) con chiavi a 1.024 bit o oltre;
 - c. [funzioni di hash](#) dei messaggi: MD5 o SHA con hash a 128 bit o oltre.
- F. Prevedere adeguati meccanismi di gestione sicura delle chiavi di cifratura. Per esempio:
 - a. meccanismi di distribuzione delle chiavi;
 - b. meccanismi di conservazione delle chiavi;
 - c. meccanismi di riciclo periodico delle chiavi;
 - d. meccanismi di revoca delle chiavi;
 - e. meccanismi di recovery delle chiavi;

- f. meccanismi di distruzione delle chiavi.
- G. Definire, in fase di design, la frequenza e le modalità del backup dei dati.
- H. Definire, in fase di design, le modalità di ripristino dei dati.
- I. Prevedere meccanismi di backup attraverso punti di sincronizzazione aperti e compatibili con le policy di backup dell'Ente.
- J. Prevedere, nel caso l'applicazione sia utilizzata per il trattamento di dati personali, meccanismi di backup dei dati con frequenza almeno settimanale.
- K. Prevedere, nel caso l'applicazione sia utilizzata per il trattamento di dati sensibili e/o giudiziari, meccanismi di ripristino dei dati in tempi non superiori ai sette giorni.
- L. Utilizzare la lista di controllo "Crittografia e disponibilità dei dati" ([Appendice B.1](#)) come strumento di supporto per verificare la rispondenza dell'applicazione a quanto richiesto nei punti precedenti.

5. Test, deployment e gestione dell'applicazione

Minacce

Indisponibilità del servizio, mancato rispetto degli obblighi previsti dalla normativa vigente, perdita di efficacia o di efficienza nel tempo delle misure di sicurezza adottate.

Contromisure

- A. Effettuare adeguati test e controlli di sicurezza sulle applicazioni prima della messa in produzione, anche in funzione del valore delle risorse da proteggere, utilizzando le liste di controllo "Test, deployment e gestione dell'applicazione" ([Appendice B.2](#)).
- B. Non utilizzare dati di produzione in ambiente di test, in particolare nel caso in cui l'applicazione sia utilizzata per il trattamento di dati sensibili e/o giudiziari.
- C. Documentare gli strumenti di gestione e di amministrazione dell'applicazione (per es. interfacce di configurazione, file di configurazione, ecc.).
- D. Effettuare, successivamente al deployment, adeguati controlli sulle applicazioni in produzione per assicurare l'efficienza e l'efficacia nel tempo dei meccanismi di sicurezza adottati.
- E. Documentare, adeguare ed aggiornare nel tempo i meccanismi di sicurezza adottati, in funzione del valore dei dati e delle informazioni da proteggere e delle minacce di sicurezza ad essi associati.

6. Requisiti minimi previsti dalla normativa vigente

Secondo quanto stabilito dalla regola 25 dell'Allegato B del D.Lgs. 196/03, qualora i destinatari del presente disciplinare siano soggetti esterni fornitori di prodotti o servizi utilizzati dall'Ente per l'adozione di misure minime di sicurezza ai sensi della normativa vigente, gli stessi devono attestare la conformità di quanto fornito alle seguenti disposizioni:

- 1) utilizzo di una procedura di autenticazione che permetta l'identificazione dell'[incaricato](#) attraverso opportune credenziali di autenticazione;

- 2) utilizzo di una parola chiave, quando prevista dal sistema di autenticazione, composta da almeno otto caratteri;
- 3) possibilità di modifica della parola chiave, quando prevista dal sistema di autenticazione, da parte dell'incaricato al primo utilizzo e, successivamente, almeno ogni sei mesi;
- 4) possibilità di disattivazione delle credenziali di autenticazione non utilizzate da almeno sei mesi, salvo quelle preventivamente autorizzate per soli scopi di gestione tecnica;
- 5) esistenza di meccanismi di autorizzazione per la separazione dei privilegi degli incaricati in base a diversi profili autorizzativi;
- 6) esistenza di meccanismi di backup che consentano il salvataggio dei dati con frequenza almeno settimanale.

Nel caso l'applicazione fornita sia destinata al trattamento di dati sensibili e/o giudiziari, l'attestazione deve inoltre indicare la conformità della stessa alle seguenti ulteriori disposizioni:

- 7) possibilità di modifica della parola chiave quando prevista dal sistema di autenticazione, da parte dell'incaricato al primo utilizzo e, successivamente, almeno ogni tre mesi;
- 8) esistenza di meccanismi di ripristino dei dati che permettano la ricostruzione degli stessi, in caso di danneggiamento, in tempi non superiori ai sette giorni;
- 9) utilizzo di tecniche di cifratura o codici identificativi, tali da rendere temporaneamente inintelligibili i dati sensibili e/o giudiziari anche a chi è autorizzato ad accedervi e da permettere l'identificazione degli **interessati** solo in caso di necessità.

L'[Appendice B.3](#) contiene una lista di controllo da utilizzare come strumento di supporto per verificare la rispondenza dell'applicazione ai requisiti minimi di sicurezza previsti dal D. Lgs. 196/03.

Appendice A - Esempio di analisi dei rischi per applicazioni web

1. Stabilire l'obiettivo di sicurezza

Rispondere ai requisiti minimi previsti dal Codice sulla protezione dei dati personali (D.Lgs. 196/03).

Garantire l'integrità dei dati, inseriti dagli utenti, successivamente alla pubblicazione sul web.

2. Descrivere sinteticamente l'applicazione

Applicazione web a tre livelli:

- *presentation*: pagine ASP.NET;
- *business logic*: class library C# + stored procedure;
- *data access*: SQL Server.

A livello infrastrutturale:

- l'applicazione è installata presso il CED della struttura
- la rete è protetta da firewall perimetrale che espone le sole porte 80 e 443
- i server (sviluppo, test e produzione) sono aggiornati periodicamente tramite sistemi di gestione automatica delle patch
- sui sistemi sono installati software antivirus aggiornati periodicamente
- le politiche di backup prevedono copia notturna dei dati di tipo incrementale

3. Minacce

Intercettazione delle credenziali di autenticazione

Intercettazione dei token o dei cookie di sessione

SQL Injection

Cross Site Scripting

4. Meccanismi di sicurezza da implementare

Autenticazione: permettere l'accesso ai dati alle sole persone autorizzate, tramite opportuna autenticazione (gli utenti si devono poter registrare sul web e la loro richiesta viene poi validata da un amministratore).

Autorizzazione: implementare due profili autorizzativi differenti: "operatore", "amministratore".

Validazione dei dati: effettuare controlli sui dati inseriti in input dall'utente e restituiti in output dall'applicazione. Utilizzare stored procedure per l'accesso ai dati di backend.

Gestione delle sessioni: prevedere meccanismi di cifratura dei token di sessione tramite funzioni di hash. Prevedere un time-out della sessione utente. Prevedere un meccanismo di forzatura del logout da parte dell'utente.

Logging: non sono previsti meccanismi di logging di sicurezza, ad eccezione dei log di

accesso riusciti/non riusciti.

Protezione dei dati: non sono previsti meccanismi di cifratura dei dati in locale. E' prevista la cifratura del traffico nella fase di autenticazione tramite protocollo SSL.

Appendice B: Liste di controllo

B.1 Design e sviluppo dell'applicazione

Analisi dei requisiti e design	
Nell'analisi dei requisiti è stato considerato il valore dei dati e delle informazioni trattate dall'applicazione	<input type="checkbox"/>
L'applicazione viene utilizzata per il trattamento di dati personali	<input type="checkbox"/>
L'applicazione viene utilizzata per il trattamento di dati sensibili e/o giudiziari	<input type="checkbox"/>
È stata eseguita l'analisi dei rischi incombenti sui dati	<input type="checkbox"/>
Sono stati considerati i vincoli architetturali e tecnologici imposti dall'infrastruttura esistente (servizi, porte, protocolli, tecnologie, ecc.)	<input type="checkbox"/>
Sono state documentate le porte ed i protocolli di comunicazione utilizzati dall'applicazione	<input type="checkbox"/>
Sono stati definiti i requisiti hardware e software necessari per il corretto funzionamento dell'applicazione	<input type="checkbox"/>
Sono stati previsti meccanismi di autenticazione degli utenti	<input type="checkbox"/>
Sono stati previsti meccanismi di autorizzazione e profilatura utenti	<input type="checkbox"/>
Sono stati previsti meccanismi di validazione dei dati in ingresso e in uscita	<input type="checkbox"/>
Sono stati previsti meccanismi di gestione sicura delle sessioni utente	<input type="checkbox"/>
Sono stati previsti meccanismi di conservazione e gestione dei log	<input type="checkbox"/>
Sono stati previsti meccanismi di disponibilità dei dati	<input type="checkbox"/>
Sono stati previsti meccanismi di cifratura dei dati	<input type="checkbox"/>

Autenticazione	
Sono stati definiti i punti di ingresso dell'applicazione che necessitano di meccanismi di autenticazione	<input type="checkbox"/>
È stato scelto il meccanismo di autenticazione considerando anche i sistemi centralizzati già parte dell'infrastruttura esistente	<input type="checkbox"/>
Sono stati rispettati i requisiti minimi obbligatori per legge nel caso di trattamento di dati personali:	
- disattivazione delle credenziali non utilizzate da almeno 180 giorni	<input type="checkbox"/>
- non riutilizzo dei codici di identificazione già impiegati assegnandoli ad altri utenti (neanche in tempi diversi)	<input type="checkbox"/>
In caso di autenticazione basata su <i>userid+password</i> , sono stati rispettati i requisiti minimi obbligatori per legge nel caso di trattamento di dati personali:	
- lunghezza minima consentita per la password di 8 caratteri	<input type="checkbox"/>
- scadenza della password non superiore ai 180 giorni (90 giorni nel caso di dati sensibili e/o giudiziari)	<input type="checkbox"/>
- possibilità per l'utente di modificare la propria password al primo login	<input type="checkbox"/>
Sono stati previsti meccanismi di lockout di un account dopo <i>n</i> tentativi di accesso non riusciti	<input type="checkbox"/>
Sono stati previsti meccanismi di delay di <i>n</i> secondi a seguito di errata	<input type="checkbox"/>

autenticazione	
Sono stati previsti meccanismi di sblocco automatico del lockout di un account dopo n minuti	<input type="checkbox"/>
Sono stati previsti meccanismi di reset e modifica delle password da parte degli utenti senza l'intervento dell'amministratore di sistema	<input type="checkbox"/>
Sono stati previsti meccanismi di cifratura delle password conservate in locale (<i>hash</i>)	<input type="checkbox"/>
Sono stati previsti meccanismi di cifratura delle password trasmesse sulla rete	<input type="checkbox"/>
Sono stati previsti meccanismi di controllo della robustezza delle password (regole di complessità)	<input type="checkbox"/>
È stato ridotto al minimo il debugging remoto, in modo da fornire all'utente, in caso di errore di autenticazione, le sole informazioni indispensabili	<input type="checkbox"/>

Autorizzazione	
Sono stati previsti meccanismi di separazione dei privilegi in funzione del profilo utente (obbligatorio nel caso di trattamento di dati personali)	<input type="checkbox"/>
I permessi sono stati assegnati secondo il principio del "minimo privilegio"	<input type="checkbox"/>
L'accesso alle risorse di sistema è limitato ai soli account privilegiati	<input type="checkbox"/>
Il login applicativo non ha accesso diretto in scrittura alle tabelle dei database	<input type="checkbox"/>
Sono utilizzate <i>stored procedure</i> per l'accesso ai database	<input type="checkbox"/>
È stata prevista la possibilità di configurare i profili autorizzativi	<input type="checkbox"/>
In caso di applicazioni web, esistono distinzioni fra aree ad accesso pubblico ed aree ad accesso riservato	<input type="checkbox"/>

Validazione dei dati	
Sono stati definiti i punti di ingresso e di uscita dell'applicazione che richiedono controlli di validazione dei dati	<input type="checkbox"/>
Sono stati previsti più livelli di controllo di validazione dei dati	<input type="checkbox"/>
I dati sono validati in termini di: <ul style="list-style-type: none"> - formato - sintassi - dimensioni 	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
Tutti i controlli di validazione sono effettuati lato server	<input type="checkbox"/>
I meccanismi di validazione sono stati scelti anche considerando gli strumenti centralizzati già disponibili	<input type="checkbox"/>
I dati sono validati in base alla strategia di: <ul style="list-style-type: none"> - accettare esclusivamente i dati riconosciuti come validi - rigettare i dati riconosciuti come invalidi - modificare i dati invalidi per renderli validi 	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
Sono stati previsti meccanismi di validazione dell'output (per es. <i>HtmlEncode</i>)	<input type="checkbox"/>
Sono stati considerati i rischi derivanti da attacchi di tipo SQL Injection	<input type="checkbox"/>
Sono stati considerati i rischi derivanti da attacchi di tipo Cross Site Scripting	<input type="checkbox"/>

Gestione delle sessioni utente	
Esistono meccanismi di time-out delle sessioni	<input type="checkbox"/>
Il contenuto degli identificatori di sessione è cifrato	<input type="checkbox"/>
Gli identificatori di sessione sono trasmessi su canali cifrati	<input type="checkbox"/>
Esistono meccanismi di logout che permettono all'utente di forzare la chiusura di una sessione	<input type="checkbox"/>

Logging	
Sono stati definiti in fase di design gli eventi chiave per la sicurezza da rilevare tramite logging	<input type="checkbox"/>
Sono previsti meccanismi di rilevamento degli eventi applicativi di: <ul style="list-style-type: none"> - autenticazione (login e logout) - accesso ai dati (lettura e scrittura) - modifica di funzioni amministrative 	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
Esistono meccanismi di modifica della granularità dei log	<input type="checkbox"/>
I log sono conservati in file a solo accesso in scrittura incrementale	<input type="checkbox"/>
I log sono conservati su supporti non riscrivibili	<input type="checkbox"/>
I log sono conservati su sistemi dedicati	<input type="checkbox"/>
Esistono meccanismi di backup dei log	<input type="checkbox"/>
Esistono meccanismi di rotazione dei log	<input type="checkbox"/>
È possibile configurare la frequenza di rotazione dei log	<input type="checkbox"/>
L'accesso ai log è consentito ai soli account privilegiati	<input type="checkbox"/>
Esistono sistemi di verifica del funzionamento dei log	<input type="checkbox"/>

Crittografia e disponibilità dei dati	
È stata eseguita un'analisi dei rischi per valutare se utilizzare o meno meccanismi di cifratura dei dati	<input type="checkbox"/>
Sono stati scelti algoritmi di cifratura standard	<input type="checkbox"/>
Sono state scelte chiavi di cifratura di lunghezza adeguata	<input type="checkbox"/>
Sono stati previsti meccanismi di firma digitale	<input type="checkbox"/>
Le chiavi di cifratura sono distinte dalle chiavi di firma	<input type="checkbox"/>
Le credenziali di autenticazione non sono salvate in chiaro	<input type="checkbox"/>
Le credenziali di autenticazione non sono trasmesse in chiaro	<input type="checkbox"/>
Sono stati previsti meccanismi di distribuzione delle chiavi	<input type="checkbox"/>
Sono stati previsti meccanismi di conservazione delle chiavi	<input type="checkbox"/>
Sono stati previsti meccanismi di riciclo periodico delle chiavi	<input type="checkbox"/>
Sono stati previsti meccanismi di revoca delle chiavi	<input type="checkbox"/>
Sono stati previsti meccanismi di recovery delle chiavi	<input type="checkbox"/>
Sono stati previsti meccanismi di distruzione delle chiavi	<input type="checkbox"/>
Sono state definite, in fase di design, le modalità e la frequenza del backup dei dati (obbligatorio nel caso di trattamento di dati personali)	<input type="checkbox"/>
Sono state definite, in fase di design, le modalità di ripristino dei dati (obbligatorio nel caso di trattamento di dati personali)	<input type="checkbox"/>
I meccanismi di backup utilizzano punti di sincronizzazione aperti e compatibili	<input type="checkbox"/>

con gli strumenti centralizzati già disponibili	
Esistono meccanismi di backup che consentono il salvataggio dei dati con frequenza almeno settimanale (obbligatorio nel caso di trattamento di dati personali)	<input type="checkbox"/>
Esistono meccanismi che consentono il ripristino dei dati in tempi inferiori ai sette giorni (obbligatorio nel caso di trattamento di dati sensibili e/o giudiziari)	<input type="checkbox"/>

B.2 Test, deployment e gestione dell'applicazione

Test	
Sono stati eseguiti i controlli indicati nelle liste di controllo sull'analisi e lo sviluppo dell'applicazione	<input type="checkbox"/>
Sono state previste verifiche nel tempo, successivamente alla fase di deployment, per assicurare il mantenimento dell'efficienza e dell'efficacia delle misure di sicurezza adottate	<input type="checkbox"/>

Deployment	
Sono stati documentati i meccanismi di sicurezza adottati	<input type="checkbox"/>
Sono state rispettate le policy centralizzate relativamente a porte, protocolli e servizi utilizzabili	<input type="checkbox"/>
È stata fornita dal fornitore esterno l'attestazione di conformità alle misure minime di sicurezza previste dalla legge	<input type="checkbox"/>

Gestione	
In ambiente di test non sono utilizzati dati di produzione	<input type="checkbox"/>
Sono stati documentati gli strumenti di gestione e di configurazione dell'applicazione	<input type="checkbox"/>
Sono stati previsti adeguati meccanismi di controllo degli accessi agli strumenti di gestione e configurazione	<input type="checkbox"/>
La connessione agli strumenti di gestione e configurazione avviene su canali cifrati	<input type="checkbox"/>

B.3 Requisiti minimi previsti dalla normativa vigente

Misure minime da osservare per tutti i trattamenti	
Esiste una procedura di autenticazione che permette l'identificazione univoca dell'utente attraverso opportune credenziali di autenticazione	<input type="checkbox"/>
È utilizzata una parola chiave (password), quando prevista dal sistema di autenticazione, composta da almeno otto caratteri	<input type="checkbox"/>
Esiste la possibilità di modifica della parola chiave, quando prevista dal sistema di autenticazione, da parte dell'utente al primo utilizzo e, successivamente, almeno ogni sei mesi	<input type="checkbox"/>
Esistono meccanismi di disattivazione delle credenziali di autenticazione non utilizzate da almeno sei mesi, salvo quelle preventivamente autorizzate per soli scopi di gestione tecnica	<input type="checkbox"/>
I codici di identificazione già impiegati non sono riutilizzati nel tempo assegnandoli ad altri utenti	<input type="checkbox"/>
Esistono meccanismi di autorizzazione per la separazione dei privilegi degli incaricati in base a diversi profili autorizzativi	<input type="checkbox"/>
Esistono meccanismi di protezione dei dati contro le minacce di intrusione e dell'azione di programmi malevoli (es. cifratura delle password, impiego di firewall o di software antivirus, hardening dei sistemi, ecc.)	<input type="checkbox"/>
I programmi sono aggiornati periodicamente per prevenire le vulnerabilità e correggerne difetti (es. patch di sistema, aggiornamenti antivirus, ecc.)	<input type="checkbox"/>
Esistono meccanismi di backup e ripristino, con salvataggio dei dati effettuato con frequenza almeno settimanale	<input type="checkbox"/>

Misure minime ulteriori da osservare nel caso di trattamenti di dati sensibili e/o giudiziari	
Esiste la possibilità di modifica della parola chiave quando prevista dal sistema di autenticazione, da parte dell'utente al primo utilizzo e, successivamente, almeno ogni tre mesi	<input type="checkbox"/>
Esistono meccanismi di ripristino dei dati che permettono la ricostruzione degli stessi, in caso di danneggiamento, in tempi non superiori ai sette giorni	<input type="checkbox"/>
Sono utilizzate tecniche di cifratura o codici identificativi, tali da rendere temporaneamente inintelligibili i dati sensibili e/o giudiziari anche a chi è autorizzato ad accedervi e da permettere l'identificazione degli interessati solo in caso di necessità	<input type="checkbox"/>

Appendice C: Glossario

Applicazione web: applicazione client/server che interagisce con l'utente (o con altri sistemi) tramite protocollo HTTP.

Bruteforce (attacco a forza bruta): tecnica di attacco per l'individuazione di una password attraverso tentativi successivi di tutte le possibili combinazioni di caratteri. Il limite degli attacchi di tipo bruteforce risiede nel tempo necessario per portarli a termine.

Buffer overflow: tecnica di attacco basata sull'alterazione del normale flusso di esecuzione di un'applicazione mediante la sovrascrittura di aree di memoria riservate.

Cross-site-scripting (XSS): tecnica di attacco che sfrutta un sito web vulnerabile per indirizzare codice maligno (iniettato dall'attaccante) verso il browser dell'utente vittima. Tale codice è eseguito sulla macchina dell'utente vittima dell'attacco.

Dati giudiziari: i dati personali idonei a rivelare provvedimenti di cui all'articolo 3, comma 1, lettere da a) a o) e da r) a u), del D.P.R. 14 novembre 2002, n. 313, in materia di casellario giudiziale, di anagrafe delle sanzioni amministrative dipendenti da reato e dei relativi carichi pendenti, o la qualità di imputato o di indagato ai sensi degli articoli 60 e 61 del codice di procedura penale.

Dati personali: qualunque informazione relativa a persona fisica, persona giuridica, ente od associazione, identificati o identificabili, anche indirettamente, mediante riferimento a qualsiasi altra informazione, ivi compreso un numero di identificazione personale.

Dati sensibili: i dati personali idonei a rivelare l'origine razziale ed etnica, le convinzioni religiose, filosofiche o di altro genere, le opinioni politiche, l'adesione a partiti, sindacati, associazioni od organizzazioni a carattere religioso, filosofico, politico o sindacale, nonché i dati personali idonei a rivelare lo stato di salute e la vita sessuale.

Denial-of-service (negazione del servizio): tecnica di attacco che limita o interrompe la disponibilità di un servizio rendendolo inaccessibile ai legittimi utilizzatori in modo temporaneo o permanente.

Funzioni di hash: funzioni matematiche che comprimono i bit di un messaggio digitale in un'impronta di dimensioni fisse (c.d. *hash*), in modo che a messaggi diversi corrispondano impronte diverse. Tali funzioni sono irreversibili, per cui dall'impronta non è possibile ricavare il messaggio digitale che l'ha originato.

Hash: impronta di un messaggio digitale calcolata tramite una funzione di hashing.

Incaricato: la persona fisica autorizzata a compiere operazioni di trattamento di dati personali.

Interessato: la persona fisica, la persona giuridica, l'ente o l'associazione cui si riferiscono i dati personali.

Meccanismo di sicurezza: elemento di un sistema progettato per preservare la confidenzialità, l'integrità e la disponibilità delle risorse del sistema stesso.

Normativa vigente: il "Codice in materia di protezione dei dati personali", Decreto Legislativo numero 196 del 30 Giugno 2003, entrato in vigore il 1° Gennaio del 2004 (c.d. "Codice della Privacy").

Privilege escalation (scalata dei privilegi): tecnica di attacco utilizzata per eseguire azioni con permessi superiori a quelli posseduti. Esempio nei sistemi Windows based: accesso di un utente appartenente al gruppo *Users* a risorse riservate al gruppo *Administrators*.

Session hijacking (dirottamento di sessione): tecnica di attacco basata sull'intercettazione di una sessione al fine di accedere a risorse senza disporre dei permessi necessari. Esempio in un'applicazione web: intercettazione di una sessione utente per accedere ad un'area web ad accesso riservato.

Spoofing (Impersonificazione): tecnica di attacco che consente ad un'identità di impersonare, in modo illegittimo, un'entità differente.

SQL Injection: tecnica di attacco che consente l'esecuzione, sul database vittima, di query SQL costruite ad-hoc a partire dall'input utente non opportunamente filtrato.

Superficie di attacco: insieme dei potenziali punti di accesso ad un sistema che possono essere sfruttati da un attaccante per comprometterne la sicurezza.