	• allarmi generati dai sistemi di controllo degli accessi; - cambiamenti o tentativi di cambiamento delle configurazioni di sicurezza del sistema. La procedura deve specificare la frequenza con cui effettuare l'audit ogni qual volta sussista la necessità e comunque non oltre il termine di 1 mese.

Rimozione delle vulnerabilità nei sistemi web	
Minaccia	- Accesso non autorizzato ai sistemi (risorse di sistema, configurazioni, interfacce amministrative, ecc.). - Accesso non autorizzato alle informazioni. - Negazione dei servizi.
Contromisure	- Il web server deve essere accuratamente testato prima che le informazioni siano rese disponibili affinché vulnerabilità e malfunzionamenti siano rimossi. - Aggiornamenti o patch rilevanti la sicurezza devono essere installati dove ritenuto necessario. Se aggiornamenti o patch sono indisponibili, devono essere adottate contromisure addizionali per la riduzione della vulnerabilità (massimizzare la difesa perimetrale) - Il web server deve essere soggetto ad una analisi delle vulnerabilità tramite strumenti automatici di vulnerability assessment e le vulnerabilità specifiche per la sicurezza ritenute critiche e riscontrate devono essere rimosse. - L'analisi delle vulnerabilità deve essere eseguita ogni qual volta sussista la necessità e comunque non oltre il termine di 1 mese.

Accordi con i fornitori di servizi internet per scoprire l'attacco	
Minaccia	- Negazione dei servizi. - Violazione della sicurezza, rispetto alle politiche di sicurezza dell'organizzazione.
Contromisure	Prevedere accordi contrattuali con il proprio Provider di servizi internet (Internet Service Provider "ISP") perché, soprattutto in fase di attacco in corso, sia possibile effettuare un'attività di tracciamento degli indirizzi di rete (IP) per tentare di individuare l'aggressore mediante un percorso a ritroso e provare a fermare l'attacco.

5.5.11 Programmazione e Configurazione

Convalida dell'input	
Minaccia	Accesso non autorizzato alle informazioni.
Contromisure	- Evitare di utilizzare parametri nella stringa di query che contengono dati sensibili o dati che possono in qualche modo influenzare la logica di protezione sul server. Utilizzare invece un identificativo di sessione per identificare il client e memorizzare sul server gli elementi sensibili nell'archivio di sessione. - Preferire l'utilizzo di HTTP POST piuttosto che HTTP GET per inviare form di dati. - Adottare un processo di serializzazione (anche noto come deflating o marshalling) in modo da convertire i dati in una sequenza di bit da trasmettere sulla rete - Crittografare i parametri passati nella query string. - Validare tutti i parametri di input al fine di garantire la conformità allo standard adottato in termini di lunghezza minima e massima consentita, range di valori numerici consentiti, sequenze di caratteri e patterns ammessi e se e quando sono consentiti valori nulli. - Utilizzare un meccanismo di Whitelisting nella validazione dei parametri di query.

Convalida dell'input	
Minaccia	Compromissione delle comunicazioni - Cross-site request forgery (CSRF)
Contromisure	- Aggiungere a ciascuna transazione un valore numerico casuale di lunghezza elevata (da usare come token). Tale valore allegato alla richiesta viene convalidato rispetto al valore dato per quella specifica sessione utente. Pertanto, un aggressore non potrà incorporare una URL che rappresenta una transazione valida nella pagina controllata dall'attaccante. Richiedere un'interazione umana aggiuntiva per le transazioni sensibili in forma di autenticazione ripetuta o risposta a CAPTCHA.

Convalida dell'input	
Minaccia	Compromissione delle comunicazioni - Manipolazione dell'intestazione HTTP
Contromisure	Non basare le decisioni di sicurezza sulle intestazioni HTTP. Ad esempio, non fidarsi di quanto riportato nell'intestazione "HTTP Referer" per determinare la provenienza di una richiesta in quanto facilmente falsificabile.

Convalida dell'input	
Minaccia	- Accesso non autorizzato alle informazioni. - Attacchi all'integrità dei sistemi - Negazione dei servizi. - (Ad es. per attacchi di injection ed esecuzione arbitraria di codice).
Contromisure	Configurare il web service in modo da garantire l'attuazione di specifici controlli dei dati in ingresso. In particolare determinare: - Il set di caratteri consentito; - La lunghezza minima e massima dei dati; - L'intervallo numerico (range) dei dati; - Quali valori sono ammessi; - Se è previsto il tipo "NULL"; - Se sono consentiti duplicati; - Il formato consentito dei nomi dei file, nel caso siano accettati come input, e verificarne la presenza nella gerarchia di directory dell'applicazione.

Convalida dell'input	
Minaccia	- Accesso non autorizzato alle informazioni. - Attacchi all'integrità dei sistemi - Negazione dei servizi. - (Ad es. con Cross-site scripting - XSS).
Contromisure	Eseguire una convalida completa dell'input. Il sistema deve garantire che l'input da query strings, i campi delle form e i cookie siano validi. Considerare tutti gli input da parte degli utenti come potenzialmente dannosi, pertanto è necessario filtrare o sanitizzare l'input lato server. Validare tutti gli input per valori validi noti e quindi rifiutare tutti gli altri input. Utilizzare espressioni regolari per convalidare i dati di input ricevuti tramite i campi di form HTML, cookie e query strings. Utilizzare le funzioni HTMLEncode e URLEncode per codificare qualsiasi output che contiene l'input dell'utente. Questo converte gli script eseguibili in HTML innocuo.

Convalida dell'input	
Minaccia	Negazione dei Servizi (Buffer overflows)
Contromisure	- Eseguire una convalida completa dell'input. Questa è la prima linea di difesa contro

	il buffer overflow. Sebbene possa esistere un bug nel processo che consente all'input atteso di sconfinare le aree di memoria allocate, gli input inattesi sono in genere la causa principale di questa vulnerabilità. Filtrare l'input convalidandolo per tipo, lunghezza, formato e range. - Quando possibile, limitare l'utilizzo di codice unmanaged (es. C, C++) da parte dell'applicazione e verificare accuratamente le API che usano codice unmanaged per garantire che l'input venga correttamente convalidato. - Esaminare i casi in cui il codice managed chiama API che usano codice unmanaged al fine di assicurare che solo parametri appropriati possano essere passati come parametri all'API. - Utilizzare specifici flag di compilazione del codice sorgente per verificare staticamente le formattazioni di stringhe e produrre eventuali avvisi di pericolo o di sospetto.

Convalida dell'input	
Minaccia	- Accesso non autorizzato alle informazioni. - Attacchi all'integrità dei sistemi - Negazione dei servizi. - (Ad es. da esecuzione arbitraria di codice).
Contromisure	- Evitate di utilizzare i nomi dei file come input dove possibile e utilizzare invece percorsi di file assoluti che non possono essere modificati dall'utente finale. Assicurarsi che i nomi dei file seguano gli standard di formato consentiti dal file system (se si deve accettare i nomi dei file come input) e convalidarli nel contesto dell'applicazione. Ad esempio, verificare che siano all'interno della gerarchia di directory dell'applicazione. - Assicurarsi che la codifica dei caratteri sia impostata correttamente per limitare il modo in cui l'input può essere rappresentato (canonicalizzazione).

Convalida dell'input	
Minaccia	Accesso non autorizzato alle informazioni (manipolazione dei campi di un Form).
Contromisure	Invece di utilizzare i campi nascosti di una form, utilizzare gli identificativi di sessione allo stato di riferimento mantenuto nell'archivio di stato lato server. Validate tutti i campi di input al fine di garantire la conformità allo standard adottato in termini di lunghezza minima e massima consentita, range di valori numerici consentiti, sequenze di caratteri e patterns ammessi e se e quando sono consentiti valori nulli. Utilizzare un meccanismo di Whitelisting nella validazione dei campi di input. Usare e configurare in modo appropriato un firewall per l'applicazione web in uso.

Convalida dell'input	
Minaccia	Accesso non autorizzato alle informazioni (HTML Injection).
Contromisure	Validare gli elementi HTML nel flusso HTTP in entrata che contiene i dati forniti dall'utente. Impiegando una convalida nativa dell'input dell'utente è possibile rimuovere qualsiasi sottostringa di sintassi HTML (come tag e link) dai contenuti testuali forniti dall'utente stesso.

Convalida dell'input	
Minaccia	Accesso non autorizzato alle informazioni (Path traversal).
Contromisure	Validare l'input proveniente dal browser web attraverso l'uso di white list. Verificare che non esistano file documentali facilmente raggiungibili o il cui percorso