

Whistleblower Software Documento sulla Sicurezza

Aggiornato Aprile 2023



Whistleblower
Software

Indice

1. Panoramica	5
2. L'architettura	5
2.1 Frontend	6
2.2 Backend	6
3. Ambiente Fisico	7
3.1 Localizzazione	7
3.2 Separazione dei dati	7
3.3 Accesso ai dati e autenticazione	7
4. Crittografia	8
4.1 In Transito	8
4.1.1 Dati in transito tra cliente e server	8
4.1.2 Dati in transito tra server e servizio	8
4.1.3 Versione TLS supportata	8
4.2 A riposo	8
4.3 Modello di crittografia personalizzato (crittografia End-2-End)	9
4.3.1 Soluzioni di crittografia supportate	9
4.3.2 Panoramica dei modelli	10
Creare un caso (prospettiva del cliente Whistleblower)	10
Aggiungere ulteriori informazioni alla segnalazione	12
Lettura dei dati (prospettiva del cliente segnalante)	12
Panoramica ad alto livello dei casi e delle chiavi	14
5. Comunicazioni e Operazioni	15
5.1 Gestione delle Patch	15
5.1.1 Servizi Gestiti e Patchati Automaticamente	15
5.1.2 Servizi patchati automaticamente	15
5.1.3 Servizi parzialmente gestiti e patchati da AWS e Whistleblower Software	15
5.2 Backup dei dati, RPO e RTO	16
5.2.1 Frequenza dei backup	16
5.2.2 Sicurezza del backup	16
5.2.3 Garanzia di cancellazione definitiva dei dati	16
5.2.4 Recovery Point Objective (RPO)	16
5.2.5 Recovery Time Objective (RTO)	16
5.3 Logging	17
5.3.1 Livelli di Logging	17
1. livello: Modifiche al sistema - Occorrenze	17
2. livello: Modifiche al sistema - Modifiche generali	17
3. livello: Applicazione - Eventi di login ecc.	18
4. livello: Applicazione - Errori di convalida	18
5. livello: Applicazione - Errori di logging	19

6. livello: Server/Applicazione - Log di flusso DNS ecc.	19
5.4 Monitoraggio dei Sistemi	20
5.5 Ciclo di vita dello sviluppo sicuro (SDLC)	20
5.5.1 Flusso del lifecycle	21
5.5.2 Principali attività del ciclo di vita eseguite	22
A. Test di penetrazione	22
B. Costruire e rilasciare a piccoli incrementi	22
C. Debito tecnico	22
D. Politica sui pacchetti di terze parti	22
E. Ricerca di fughe di segreti	23
F. Controllo versione	23
G. Controllo delle vulnerabilità	23
H. Standard di crittografia	23
I. Metriche e logging	23
J. Formazione degli sviluppatori	23
K. Testing	24
L. Separazione degli ambienti	24
6. Gestione dell'Accesso e dell'Autenticazione	25
6.1 Accesso alle segnalazioni nella piattaforma	25
6.2 Provisioning degli utenti e le loro autorizzazioni	26
6.3 Fornitori di login disponibili	26
6.3.1 Email e password	26
6.3.2 OAuth 2.0	27
6.3.3 SAML 2.0	27
7. Gestione delle vulnerabilità	28
7.1 Evitare le vulnerabilità	28
7.1.1 Superficie limitata:	28
7.2 Scansione delle vulnerabilità	29
I. Scansioni di vulnerabilità delle dipendenze del codice	29
II. Scansione della vulnerabilità del contenitore/server	29
III. Scansione di vulnerabilità dell'infrastruttura/rete	30
8. Compliance, certificazione e audit	31
8.1 ISAE 3000 Tipo 2	31
8.2 ISO 27001	31
8.3 Test di penetrazione	31
9. Caratteristiche di Sicurezza dell'Account	32
9.1 Periodo di conservazione fisso	32
9.2 Whitelist degli indirizzi IP per il login dell'amministratore	32
9.3 Autenticazione a più fattori	32
9.4 Liste di controllo degli accessi (ACL)	32
9.5 Single Sign-On (SSO)	32

9.5.1 OAuth 2.0	33
9.5.2 SAML 2.0	33
10. Protezione dell'Anonimato	34
10.1 Nessuna memorizzazione dell'indirizzo IP o dell'ID hardware del segnalante	34
10.2 Rimozione dei metadati	34
10.3 Distorsione della voce	34

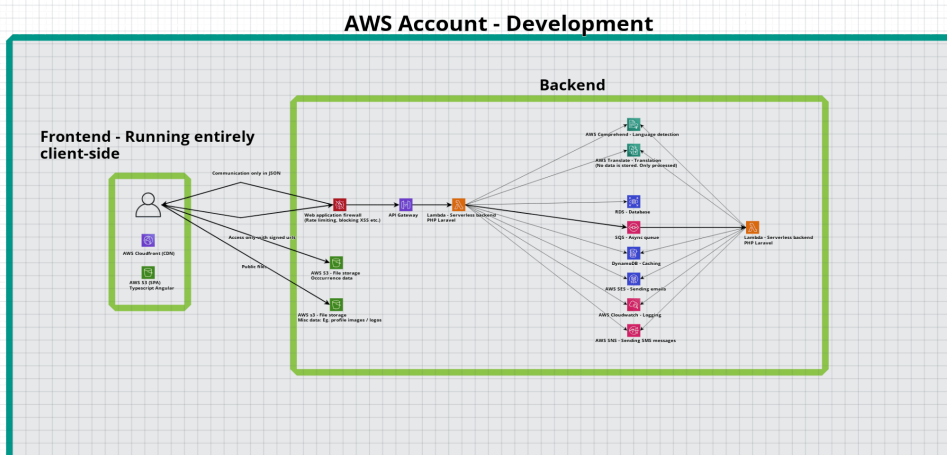
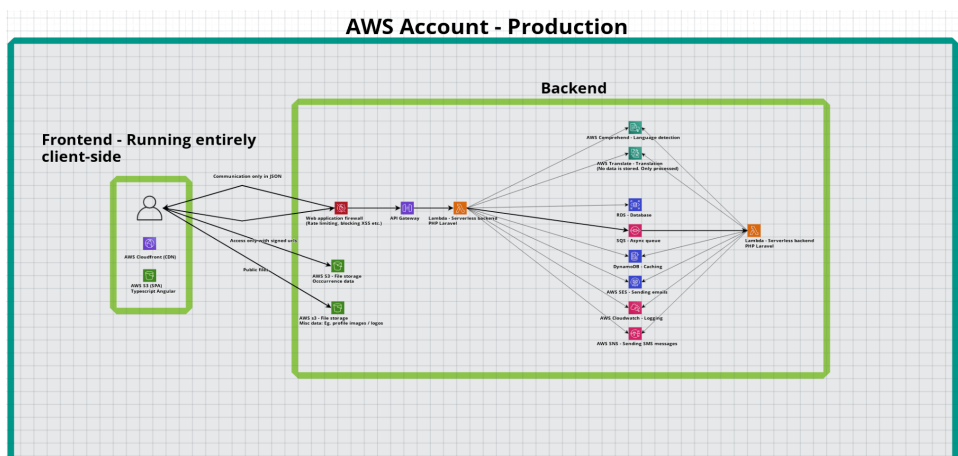
1. Panoramica

Whistleblower Software ApS fornisce una soluzione tecnica che aiuta aziende e consulenti esterni nella gestione di canali di segnalazione che siano user-friendly e sicuri. Questo documento descrive le misure tecniche legate alla sicurezza.

2. L'architettura

Qui trovate uno schema della configurazione. Nota:

- SPA e API sono separati.
- Account AWS separati per ogni ambiente
- Alcuni servizi potrebbero non essere elencati nel diagramma, come AWS Config (che registra le modifiche all'infrastruttura).
- La maggior parte dell'infrastruttura è gestita e modificata tramite codice. L'infrastruttura come codice (Terraform) viene conservata e le modifiche vengono registrate nel sistema di versionamento del codice, Github.



2.1 Frontend

Il frontend è sviluppato come applicazione a pagina singola (SPA). Questo porta diversi vantaggi, ma il benefit principale è la netta separazione dal backend.

Il framework utilizzato per la SPA è Angular, scritto in Typescript (compilato in Javascript).

Il frontend è ospitato su AWS S3, con un'istanza di CloudFront davanti al bucket.

2.2 Backend

Il backend è sviluppato come un' API REST JSON, che è consumata dalla SPA frontend. Il framework utilizzato per l'API è Laravel, scritto in PHP.

L'API viene eseguita serverless su AWS Lambda, distribuito come container docker. Il backend utilizza una vasta gamma di servizi da AWS per completare diverse attività, come la traduzione di testi (se abilitata), l'invio di e-mail o SMS e l'hosting di file.

** Nota che i nostri server web sono contenitori in sola lettura (ad eccezione della directory /tmp), che vengono completamente reimpostati con una nuova istanza di container ogni 15 minuti o in caso di distribuzioni. Questo limita la capacità del malware di persistere in generale, ma anche tra patch e distribuzioni.*

3. Ambiente Fisico

3.1 Localizzazione

Tutti i dati sono archiviati con AWS nel data center di Francoforte, in Germania. I backup sono archiviati in diverse zone di disponibilità.

3.2 Separazione dei dati

Whistleblower Software conserva i dati del cliente in due diversi sistemi (entrambi sono servizi gestiti, il che minimizza i rischi):

- AWS RDS (Database - archiviazione dei dati del caso come titolo, descrizione, ecc. del caso)
 - I dati del caso sono crittografati E2E (oltre alla crittografia at-rest)
- AWS S3 (File system - archiviazione dei file)
 - I dati del caso sono crittografati E2E (oltre alla crittografia dei dati at-rest)
 - I dati del caso sono archiviati in un file system diverso rispetto ai file vari (immagini del profilo, ecc.).

3.3 Accesso ai dati e autenticazione

L'autorizzazione avviene caso per caso. Ciò significa che solo le persone che hanno accesso al caso potranno recuperare i dati.

Su AWS S3 tutti i file sono archiviati in un bucket privato senza accesso pubblico. L'accesso temporaneo al file richiesto viene concesso se un utente ha accesso al file. Tutte le richieste di file (per visualizzazione o per scaricare il documento) vengono registrate. Su AWS RDS è possibile accedere ai dati solo tramite nostre API. Questo garantisce che solo gli utenti con accesso al caso specifico possano leggerne i dati.

Entrambi questi sistemi hanno tutti i dati dei casi crittografati E2E. Ciò implica che gli unici detentori della chiave privata sono le amministrazioni aggiudicatrici e che i dati dei casi non possono essere letti da nessun altro.

** AWS garantisce che tutti i dati vengano cancellati da qualsiasi hardware nel caso di un potenziale riutilizzo dello spazio memorizzato. Oltre a essere cancellati prima di un nuovo utilizzo, tutti i nostri dati sono crittografati at rest (e crittografati E2E), quindi anche nel caso in cui non siano stati cancellati, sono comunque illeggibili.*

4. Crittografia

Tutti i dati vengono crittografati sia quando sono in transito che a riposo. Questo assicura l'*integrità dei dati*, in quanto i dati non possono essere corrotti o modificati durante il trasferimento, la *privacy*, in quanto i dati non possono essere intercettati da terze parti, e l'*autenticazione*, in quanto l'utente finale può essere certo che il sito a cui si collega sia effettivamente Whistleblower Software ApS.

4.1 In Transito

4.1.1 Dati in transito tra cliente e server

I dati in transito vengono crittografati utilizzando il protocollo TLS (Transport layer security), questo viene eseguito dal browser dell'utente finale e dai nostri server

Informazioni aggiuntive sul protocollo HTTPS:

- Tutte le richieste HTTP vengono forzatamente reindirizzate a HTTPS. Nessuna richiesta viene gestita via HTTP.
- Tutte le richieste che utilizzano versioni di TLS inferiori a 1.2 vengono bloccate.
- Headers HSTS sono sempre impostati, obbligando caricamenti solo tramite HTTPS. Tutti i nostri domini sono elencati nell'elenco precaricato HSTS, che viene scaricato con tutti i principali browser.

4.1.2 Dati in transito tra server e servizio

Nella maggior parte dei casi, quando alcuni dati vengono inviati al server dal cliente, è necessario poi memorizzarli da qualche parte. Quando ciò accade, i dati si spostano tra un server web e ad es. un database. Tutte le comunicazioni interne al server sono crittografate con TLS. Questo include code, database, cache, ecc.

4.1.3 Versione TLS supportata

Sono supportate solo le versioni di TLS uguali o superiori a 1.2; tutte le altre versioni vengono bloccate.

4.2 A riposo

Ogni volta che i dati vengono archiviati su un disco rigido, come ad esempio in un database, questi vengono crittografati utilizzando AES (256 bit). Questo impedisce la lettura dei dati in caso di furto fisico di un disco rigido in un data center.

4.3 Modello di crittografia personalizzato (crittografia End-2-End)

Oltre all'uso della crittografia di cui sopra citato, noi di Whistleblower Software abbiamo sviluppato un modello di crittografia end-to-end personalizzato. Ciò significa che **prima che i nostri server ricevano** qualsiasi dato, tutti i dati rilevanti del caso vengono crittografati sul lato client (o sul dispositivo del segnalante o su quello del gestore del caso). Ciò significa che i dati vengono crittografati prima ancora che entrino in gioco l'HTTPS e la normale crittografia del disco.

Quando una segnalazione viene inviata, è generato sul lato cliente un AES-GCM con una chiave di lunghezza di 256 bit e un tag di 128 bit. Tutte le informazioni vengono quindi crittografate dal cliente utilizzando questa chiave.

Il modello è implementato utilizzando [Web Crypto API](#) e [phpseclib](#) utilizzando una combinazione di:

1. AES-GCM con lunghezza chiave di 256 bit (lunghezza tag 128 bit) - Algoritmo di crittografia simmetrica
2. RSA con lunghezza di chiave di 4096 bit - Algoritmo di crittografia asimmetrica
3. PBKDF2 (400.000 iterazioni e una lunghezza del salt di 128 bit) - Algoritmo di hashing

Ogni volta che una segnalazione o i documenti allegati a questa vengono letti da qualcuno, una versione crittografata viene recuperata dal server. Quando il contenuto viene recuperato, questo viene decifrato utilizzando la chiave AES del caso. (La chiave AES del caso viene decifrata utilizzando la chiave privata dell'azienda/partner/segnalante).

4.3.1 Soluzioni di crittografia supportate

Nella nostra soluzione sono attualmente disponibili due opzioni per i nostri clienti:

1. Gestiamo una password principale che viene utilizzata per decifrare la chiave privata RSA.
2. Il partner/l'azienda gestisce una password principale che viene utilizzata per decifrare la propria chiave privata RSA (in combinazione con i segnalanti che gestiscono la propria password principale per la chiave privata RSA, si ottiene la crittografia end-to-end).

Le chiavi private RSA sono sempre crittografate con AES-GCM con una lunghezza della chiave di 256 bit e una lunghezza del tag di 128 bit. La chiave AES viene generata utilizzando una password principale di almeno 16 caratteri che viene sottoposta a hashing pbkdf2 con 400.000 iterazioni e una lunghezza del salt di 128 bit.

Non è consentito decifrare manualmente nessuno dei campi crittografati nel database nel caso in cui gestiamo la password principale del cliente.

4.3.2 Panoramica dei modelli

A. Creare un caso (prospettiva del cliente Whistleblower)

1. Una **chiave AES del caso** viene generata dal lato client. (1 chiave AES per caso).
2. Tutti i dati del caso vengono crittografati utilizzando la **chiave AES del caso**.
3. La **chiave AES del caso** viene crittografata come descritto:
 - a. Utilizzando una **chiave RSA Pubblica dell'Azienda**. (1 coppia di chiavi RSA per azienda)
 - b. Utilizzando una **chiave RSA Pubblica del Segnalante** generata dal lato client. (1 coppia di chiavi RSA per segnalante)
4. La **chiave privata** per la **chiave RSA Pubblica del Segnalante** è crittografata utilizzando una **chiave AES derivata dall'hashing PBKDF2**.

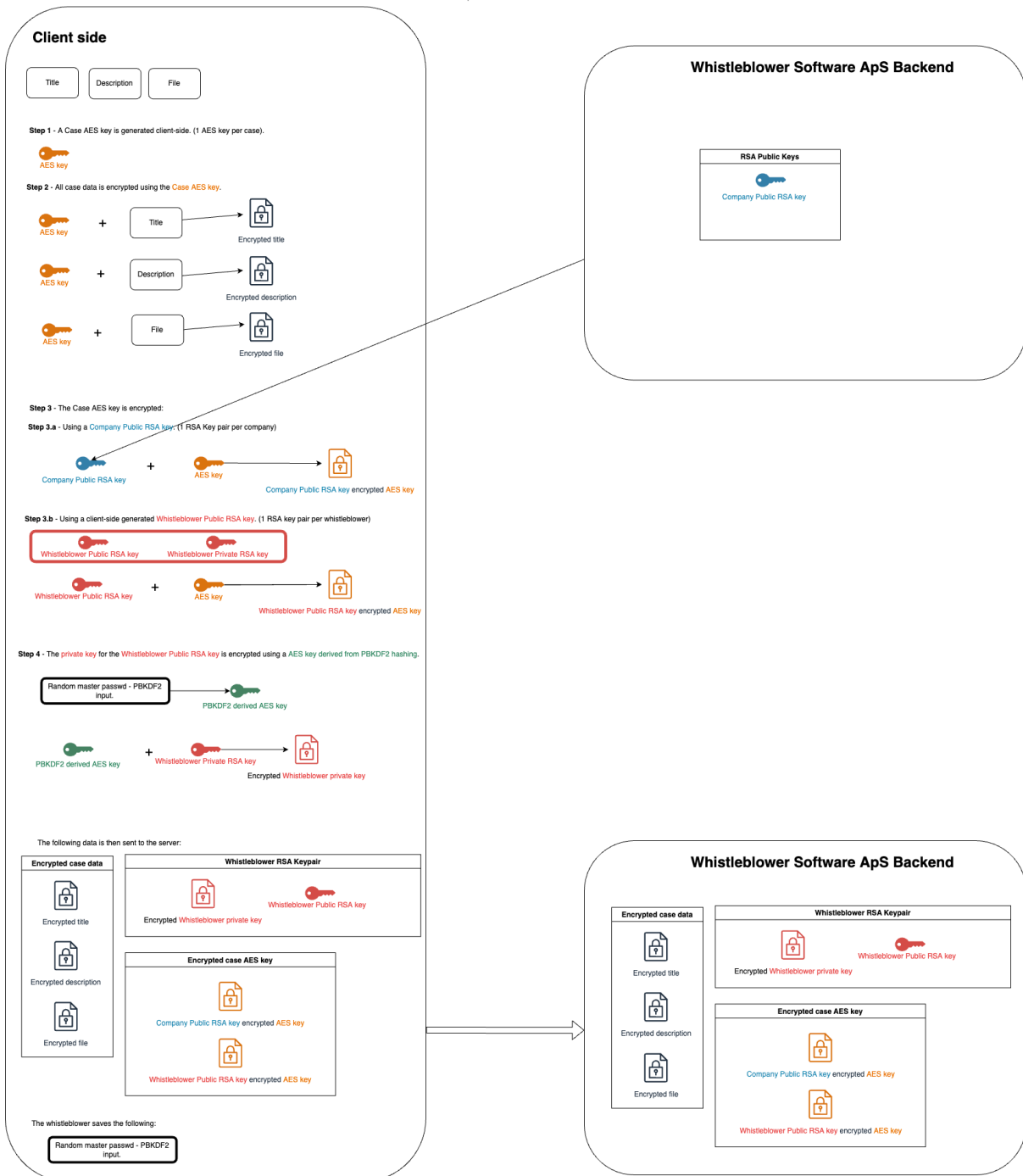
I seguenti dati vengono poi inviati al server:

- Dati del caso crittografati
- **Chiave RSA Pubblica dell'Azienda** crittografata **Chiave AES**
- **Chiave RSA Pubblica del Segnalante** crittografata **Chiave AES**
- **Chiave privata** crittografata con **chiave AES derivata da PBKDF2** del segnalante

Il segnalante salva quanto segue:

- L'input **PBKDF2** (una password casuale di 16 caratteri generata dal lato client).

Creating a case (Whistleblower client perspective):



Vedi [qui](#) il diagramma in grandezza naturale.

B. Aggiungere ulteriori informazioni alla segnalazione

Quando vengono aggiunti dati alla segnalazione, questi vengono crittografati utilizzando la stessa **chiave AES** generata all'inizio. Questo significa che se è possibile leggere i dati della segnalazione, è anche possibile aggiungere nuovi contenuti.

C. Lettura dei dati (prospettiva del cliente segnalante)

I seguenti dati vengono recuperati dal server:

- Dati del caso crittografati
- **Chiave RSA pubblica del Segnalante** crittografata **Chiave AES**
- **Chiave privata** del segnalante crittografata con **chiave AES derivata da PBKDF2**

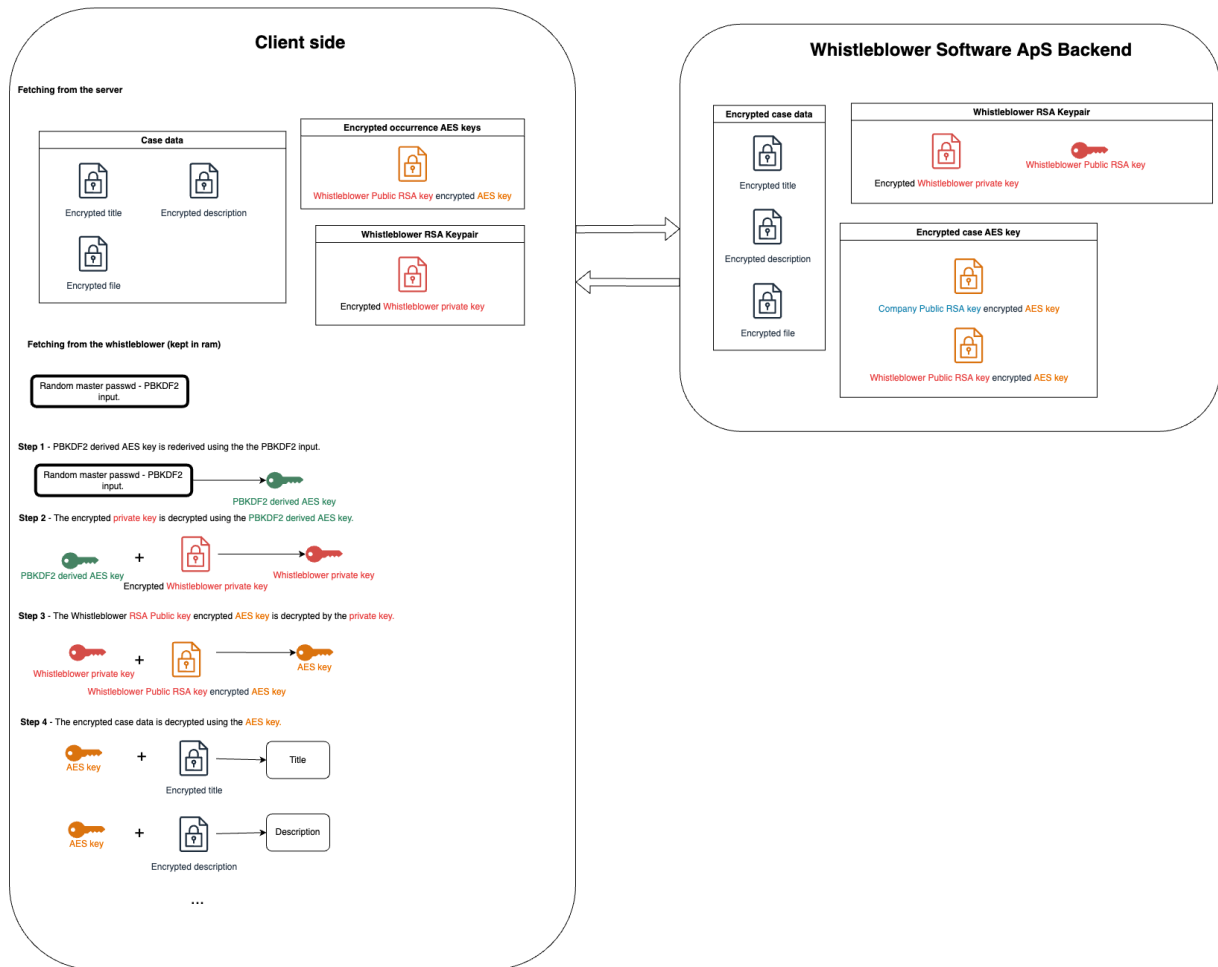
I seguenti dati sono recuperati dal segnalante:

- L'input **PBKDF2** (una password casuale di 16 caratteri generata dal lato client).

I dati sono adesso decifrati:

1. La **chiave AES derivata da PBKDF2** viene nuovamente ottenuta utilizzando l'input **PBKDF2**.
2. La **chiave privata** crittografata viene decifrata utilizzando la **chiave AES derivata da PBKDF2**.
3. La chiave AES crittografata con la **chiave pubblica RSA** del segnalante viene decifrata dalla **chiave privata**.
4. I dati crittografati del caso vengono decifrati utilizzando la **chiave AES**.

Reading the data (Whistleblower client perspective)

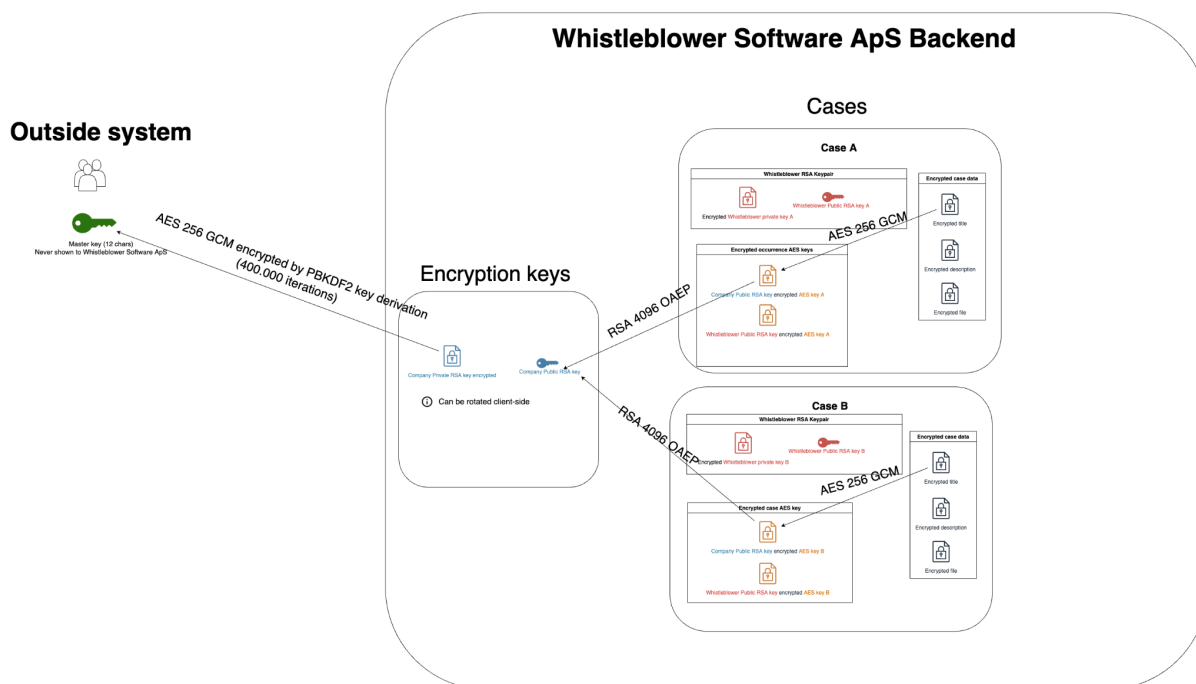


Vedi il diagramma a [grandezza naturale qui](#).

D. Panoramica ad alto livello dei casi e delle chiavi

Tutte le chiavi crittografate finiscono per essere criptate con una *master key*. Questa master key può essere esportata dal nostro sistema o potete scegliere di lasciare che sia Whistleblower Software a gestire la master key per voi.

Il diagramma seguente mostra le chiavi e i dati nel sistema quando vengono creati due casi per un'azienda:



Vedi il diagramma [a grandezza naturale qui](#).

5. Comunicazioni e Operazioni

5.1 Gestione delle Patch

Noi di Whistleblower Software cerchiamo di utilizzare il più possibile i servizi gestiti da AWS, che garantiscono un'elevata disponibilità e servizi molto sicuri. Tuttavia, vi sono servizi la cui gestione non può essere presa in carico da AWS.

5.1.1 Servizi Gestiti e Patchati Automaticamente

La maggior parte dei servizi è gestita e patchata da AWS, garantendo un'elevata disponibilità e un servizio molto sicuro. Esempi di servizi sono:

- File system
- Database
- Networking
- Sistema di coda

Le patch vengono scansionate e applicate continuamente.

5.1.2 Servizi patchati automaticamente

Alcuni servizi non sono gestiti da AWS ma vengono patchati automaticamente. Ad esempio:

- Jump box

Bonus: il nostro jump box si trova all'interno di una rete privata e non è possibile accedervi senza l'autenticazione a due fattori. Tutti i logs vengono registrati.

Le patch vengono scansionate e applicate continuamente.

5.1.3 Servizi parzialmente gestiti e patchati da AWS e Whistleblower Software

Alcuni dei nostri servizi non sono gestiti da AWS, questi sono scansionati e patchati continuamente.

- Servizi web
- Rimozione dei metadati

5.2 Backup dei dati, RPO e RTO

5.2.1 Frequenza dei backup

Vengono effettuati backup periodici per garantire che i dati dei clienti non vadano persi. I backup vengono eseguiti con i seguenti intervalli:

- Backup giornaliero che scade dopo 35 giorni.
- Backup settimanale che scade dopo 85 giorni.

5.2.2 Sicurezza del backup

Tutti i backup sono archiviati in diverse zone di disponibilità per evitare perdite di dati. Tutti i backup sono protetti secondo gli stessi standard dell'ambiente di produzione.

5.2.3 Garanzia di cancellazione definitiva dei dati

Per garantire che la possibilità di cancellazione definitiva dei dati sia in conformità al GDPR, memorizziamo tutte le queries di cancellazione definitiva ed eliminiamo nuovamente le righe cancellate con una migrazione nel caso di ripristino di backup.

Nel caso in cui si debbano ripristinare i dati da un backup, si seguirà questa procedura:

1. Ripristino del database senza accedervi.
2. Esecuzione di una migrazione per cancellare tutti i dati eliminati in precedenza, a partire dai dati raccolti sui dati eliminati in precedenza (si noti che tutte le richieste di eliminazione vengono anche queste memorizzate in più ambienti, in modo che in caso di rollback i dati siano ancora a portata di mano).
3. Disponibilità di accesso al database.

5.2.4 Recovery Point Objective (RPO)

Eseguiamo backups come sopra indicato. Una volta al giorno. Ciò significa che una potenziale perdita massima sarebbe di 23 ore e 59 minuti.

Tuttavia, oltre a questi backup definitivi, abbiamo la possibilità di eseguire un rollback ad un punto preciso con un delay di 5 minuti. Questo non è sempre possibile, ma lo è nella maggior parte degli scenari. In questo caso, la perdita massima sarebbe quasi nulla.

5.2.5 Recovery Time Objective (RTO)

Testiamo periodicamente i nostri backups e la nostra capacità di ripristinare la perdita di sistemi. Il nostro obiettivo è mantenere il Recovery Time Objective sotto le 4 ore.

5.3 Logging

5.3.1 Livelli di Logging

Whistleblower Software prende sul serio il logging, questo è esteso in tutta l'applicazione, migliorandola continuamente.

Registriamo i seguenti livelli:

1. livello: Modifiche al sistema - Occorrenze

Il primo livello di logging utilizzato da Whistleblower Software è la registrazione di ciò che è accaduto a un evento. Questo primo livello di registrazione è visibile agli utenti del sistema, in modo che consulente e utente possano vedere le rispettive modifiche a livello della segnalazione.

Esempi di logging possono essere:

- Chi ha visto un evento, quando.
- Chi ha risposto a un evento, quando e come.
 - Ad esempio, "Luca ha inviato un messaggio al segnalante che dice 'Grazie per la sua segnalazione. Approfondiremo la vicenda il prima possibile.'" (2 giugno 2021 20.56)
- Quali valori sono cambiati, da cosa a cosa, quando e da chi.
 - Ad esempio, "Luca ha modificato il livello di *gravità* da 2 a 5" (2 giugno 2021 21.23)
- Un registro completo di chi ha avuto accesso a un caso, per quanto tempo e chi ha dato/rimosso loro l'accesso.
 - Ad esempio, "Luca ha dato accesso all'evento a Giovanna, Marco e Andrea" (3 giugno 2021 09.06)

Questi registri sono disponibili caso per caso, solo per gli utenti che hanno accesso alla segnalazione specifica.

Gli utenti e i consulenti hanno accesso ai registri solo in modalità di lettura. Ciò significa che i registri non possono essere modificati/cancellati dagli utenti del sistema.

2. livello: Modifiche al sistema - Modifiche generali

Il secondo livello di registri utilizzati da Whistleblower Software è la registrazione di ciò che è accaduto al sistema nel suo complesso. Tutte le autenticazioni e gli eventi di sicurezza sono disponibili direttamente nel sistema. Altri logs di secondo livello sono disponibili su richiesta.

Esempi di logging possono essere:

- Modifiche apportate agli utenti:
 - "Luca ha cambiato il nome dell'utente 'Giovanna', il suo nuovo nome 'Andrea'" (4 giugno 2021, 12.36)
 - "Luca ha cancellato il seguente utente: 'Giovanna'" (4 giugno 2021, 12.46)
- Modifiche apportate alle impostazioni del sistema:
 - "Luca ha disabilitato la possibilità di creare una segnalazione con categoria 'Sessismo'".

Utenti e consulenti non hanno accesso a questi registri. I registri non possono essere modificati/cancellati dagli utenti del sistema.

3. livello: Applicazione - Eventi di login ecc.

Il terzo livello di registri utilizzati da Whistleblower Software è il logging di eventi personalizzati. Anche questi eventi sono disponibili su richiesta.

Esempi di logging possono essere:

- Eventi di login
 - "Luca si è collegato dall'IP xx.xxx.xx.xxx con agente utente xxxxx". ****(4 giugno 2021, 12.46)
 - "Luca ha aggiornato la sessione dall'IP xx.xxx.xx.xxx con agente utente xxxxx". ****(4 giugno 2021, 13.06)

Gli utenti e i consulenti non hanno accesso a questi registri. I registri non possono essere modificati/cancellati dagli utenti del sistema.

4. livello: Applicazione - Errori di convalida

Il quarto livello di registri utilizzato da Whistleblower Software è il logging degli errori di convalida delle applicazioni.

Esempi di logging possono essere:

- Logica aziendale
 - "Luca ha tentato di creare un nuovo utente con un'email già utilizzata" (4 giugno 2021, 13.06)
- Tentativi di uso di campi non validi/in eccesso
 - "Luca ha cercato di creare un utente con l'email 'email-non-esistente'" (4 giugno 2021 13.06)
 - "Luca ha cercato di creare un utente con il nome 16 mb". (4 giugno 2021, 13.06)

Questi risultati vengono esaminati periodicamente dagli sviluppatori di Whistleblower Software.

5. livello: Applicazione - Errori di logging

Il quinto livello di registrazione utilizzato da Whistleblower Software è il logging degli errori di Applicazione. Si tratta di errori generati nel codice quando si verifica qualcosa di inaspettato. (Molto raro)

Esempi di logging possono essere:

- Si è verificato un errore durante il tentativo di disabilitare una categoria.

Questi registri non sono disponibili a nessuno, tranne che per il team di Whistleblower Software.

Questi risultati vengono esaminati e classificati come prioritari dagli sviluppatori di Whistleblower Software. (Uno sviluppatore viene informato immediatamente dopo il verificarsi dell'errore).

6. livello: Server/Applicazione - Log di flusso DNS ecc.

Il sesto livello di registri utilizzati da Whistleblower Software è il logging del comportamento di Server / Applicazione. Questi registri vengono analizzati automaticamente da intelligenza artificiale, rilevamento delle anomalie e intelligence integrata per minacce.

I dati analizzati comprendono:

- Log di Flusso VPC
- Log DNS
- Log delle modifiche al sistema di archiviazione dei dati
 - Ad esempio, i file delle impostazioni disponibili pubblicamente, ecc.
- Eventi di gestione del servizio in hosting
 - Tentativo di accesso al provider di hosting riuscito/non riuscito, modifiche al provisioning del server/modifiche alle impostazioni del server
 - Tentativo di forza bruta

Questi risultati vengono rivisti periodicamente dagli sviluppatori di Whistleblower Software.

I registri e i risultati sono di sola lettura, il che significa che nessuno è in grado di alterare/modificare i registri e i risultati.

5.4 Monitoraggio dei Sistemi

I nostri sistemi sono regolarmente monitorati con l'aiuto di soluzioni gestite e personalizzate per garantire un'elevata disponibilità e un sistema sicuro.

Con l'aiuto dei servizi gestiti, monitoriamo continuamente il flusso di dati nel nostro network tra i nostri sistemi e veniamo avvisati in caso di attività sospette (meglio conosciuto come Network Intrusion Detection System). Inoltre, l'attività dell'amministratore viene registrata e analizzata per individuare errori e potenziali abusi. Per saperne di più, leggete qui:

- <https://aws.amazon.com/guardduty/>

Un'altra misura che abbiamo implementato per monitorare e proteggere i nostri sistemi è il Web Application Firewall. Il WAF è una misura aggiuntiva che aiuta a prevenire e monitorare i tentativi di hacking. Per saperne di più, leggete qui:

- <https://aws.amazon.com/waf/>

Oltre ai sistemi gestiti, abbiamo implementato sistemi personalizzati per rilevare i tempi di inattività e per individuare gli errori che si verificano nell'applicazione.

La soluzione comprende:

- Rilevamento dei tempi di inattività e delle API/frontend che non rispondono a 200 OK.
- Individuazione e registrazione di tutti gli errori che si verificano.
- Lo stato è visibile a questo link: <https://stats.uptimerobot.com/1EYx1Cp76n>

Il rilevamento di intrusioni fisiche e molte altre misure garantiscono una sicurezza fisica ai massimi livelli. Per saperne di più sul monitoraggio fisico e altro ancora, cliccate qui:

- <https://aws.amazon.com/compliance/data-center/>

5.5 Ciclo di vita dello sviluppo sicuro (SDLC)

La sicurezza è una priorità assoluta per Whistleblower Software. Un ciclo di vita dello sviluppo sicuro (SDLC) è un processo che ci aiuta a garantire che la sicurezza sia integrata nel software fin dall'inizio.

Il nostro SDLC comprende varie attività di sicurezza che vengono eseguite in ogni fase del processo di sviluppo. Queste attività ci aiutano a trovare e risolvere potenziali vulnerabilità di sicurezza e a prevenire l'introduzione di nuove.

5.5.1 Flusso del lifecycle

Il nostro obiettivo è fare in modo che ogni fase del ciclo di vita di sviluppo sia il più sicuro ed efficiente possibile, dalla pianificazione fino alla manutenzione. Per questo abbiamo adottato molte misure per rendere il nostro ciclo di vita di sviluppo il più ottimale possibile.

Il nostro flusso attuale è il seguente:

1. Pianificazione e analisi di nuove funzionalità in collaborazione con i clienti.
2. Design e implementazione su un ramo di funzionalità separato in GIT.
3. Si raccolgono i potenziali feedback dei clienti e si ripetono 1-2.
4. I test, l'analisi statica del codice, la complessità ciclomatica, il linting e ulteriori controlli vengono effettuati prima di passare al ramo di funzionalità.

5. Viene fatta una richiesta di pull dal ramo di funzionalità al ramo di sviluppo, dove vengono eseguiti nuovamente tutti i controlli.
6. Viene effettuata una revisione del codice da parte di uno sviluppatore principale.
 - 1) In questa fase vengono eseguite attività fondamentali come i controlli sui pacchetti di terze parti, gli standard di crittografia, ecc.
 - 2) Se è necessario un feedback o una modifica da parte dello sviluppatore principale, i passaggi da 2 a 6 vengono ripetuti fino all'approvazione delle modifiche
7. Le modifiche vengono ora unite e distribuite automaticamente nell'ambiente di sviluppo, dove le modifiche allo schema o la potenziale manipolazione dei dati vengono testate con i dati di prova.
8. I test manuali e il controllo qualità vengono eseguiti nell'ambiente di sviluppo.
9. Viene fatta una richiesta di pull dal ramo di sviluppo al ramo master. Vengono eseguiti nuovamente tutti i controlli (test, analisi statica del codice, ecc.).
10. In caso di modifiche allo schema o di manipolazione dei dati, il test viene eseguito su un mirror **temporaneo** del database di produzione.
11. Le modifiche vengono unite e distribuite automaticamente nell'ambiente di produzione e vengono eseguite eventuali modifiche allo schema o manipolazioni dei dati.
12. A seconda delle modifiche, il test manuale delle modifiche distribuite viene testato anche in produzione.
13. Le metriche chiave, come le eccezioni e gli errori, vengono registrate e monitorate per migliorare la sicurezza e le prestazioni.

In qualsiasi fase, siamo in grado di eseguire un rollback a una versione precedente della piattaforma.

5.5.2 Principali attività del ciclo di vita eseguite

A. Test di penetrazione

Il test di penetrazione è un tipo di test di sicurezza che prevede il tentativo di attaccare il nostro software per individuare le vulnerabilità di sicurezza. Questo test viene eseguito annualmente da un contractor indipendente di terze parti. I risultati sono disponibili su richiesta dopo che i potenziali problemi sono stati risolti.

B. Costruire e rilasciare a piccoli incrementi

Stiamo costruendo e rilasciando a piccoli incrementi, in modo da poter ottenere feedback e rivedere i potenziali problemi di sicurezza in tempo e con frequenza. In questo modo, possiamo ridurre il rischio che le vulnerabilità di sicurezza vengano distribuite in produzione e che diventino critiche o addirittura invalidanti.

Questo garantisce anche che le revisioni del codice non si accumulino fino a diventare non revisionabili e che la velocità di sviluppo non si arresti bruscamente.

C. Debito tecnico

I rischi di sicurezza del debito tecnico sono spesso sottovalutati. Se non ci si occupa del debito tecnico, si incorre in un rischio aggiuntivo che può portare a vulnerabilità di sicurezza. Ecco perché noi di Whistleblower Software ci stiamo concentrando sulla riduzione del debito tecnico per essere in grado di muoverci velocemente e rilasciare rapidamente features.

D. Politica sui pacchetti di terze parti

La sicurezza del nostro software è importante per noi, e parte di questa sicurezza consiste nel garantire che i pacchetti open source che utilizziamo siano sicuri. Per questo motivo abbiamo adottato una politica che prevede che tutti i pacchetti open source siano sottoposti a una verifica di sicurezza prima di poter essere utilizzati.

E. Ricerca di fughe di segreti

Esaminiamo continuamente il nostro controllo versione per individuare potenziali fughe di dati riservati.

Molte violazioni della sicurezza avvengono a causa di fughe di dati. E di solito questi dati sono memorizzati in sistemi di controllo delle versioni come Git. Eseguiamo una scansione e cerchiamo di individuare i segreti che sono stati accidentalmente commessi.

F. Controllo versione

Archiviamo tutto il nostro codice e le modifiche apportate su GitHub. Ciò consente ai nostri sviluppatori di tenere traccia delle modifiche apportate al code base e di tornare alle versioni precedenti, se necessario. Il nostro development flow su Github assicura che i test, lo stile del codice e l'analisi statica del codice vengano eseguiti tra ogni piccola modifica al code base e che tutte le modifiche vengano riviste da un altro sviluppatore prima di essere unite. Questo processo ci aiuta a identificare e risolvere tempestivamente potenziali vulnerabilità di sicurezza.

G. Controllo delle vulnerabilità

Esaminiamo continuamente i nostri sistemi e pacchetti alla ricerca di vulnerabilità di sicurezza e interveniamo per correggere quelle identificate.

H. Standard di crittografia

Poiché utilizziamo e sviluppiamo su una piattaforma crittografata end-to-end, una parte del nostro ciclo di vita di sviluppo consiste nell'esaminare e approvare nuovi modelli di crittografia da utilizzare. Gli incrementi contenenti queste modifiche/aggiunte vengono esaminati e approvati da uno sviluppatore principale in base a una serie di standard di crittografia.

I. Metriche e logging

Come parte delle operazioni e della manutenzione del nostro sistema, vengono monitorate e raccolte una serie di metriche chiave come il tempo di attività, le eccezioni e lo stato HTTP. Questi dati vengono utilizzati per migliorare la stabilità, la sicurezza e le prestazioni del nostro sistema.

J. Formazione degli sviluppatori

Incoraggiamo i nostri sviluppatori a partecipare a eventi incentrati sulla sicurezza, come gli incontri OWASP. Questo non solo li aiuta a rimanere aggiornati sulle ultime minacce, ma li rende anche più attenti alla sicurezza durante la programmazione. I nostri sviluppatori hanno constatato che questo tipo di eventi sono vitali per mantenere le loro competenze aggiornate.

K. Testing

Sin dagli inizi di Whistleblower Software, i test sono stati una priorità assoluta, garantendo la nostra capacità di procedere rapidamente senza compromettere le funzionalità.

Abbiamo implementato un'ampia gamma di test, tra cui:

- **Test unitari:** si concentrano su piccole porzioni di codice, o unità, e testano come funzionano insieme.
- **Test di integrazione:** si basano sui test delle unità e verificano il funzionamento delle unità in quanto sistema.
- **Test end-to-end:** si concentrano sulla simulazione di reali scenari utente per garantire che l'applicazione funzioni come previsto dall'inizio alla fine.

Questo ci permette di testare sia le nuove funzionalità che le correzioni di bug prima che vengano distribuite sui nostri server di staging e di produzione.

Come parte del nostro ciclo di vita di sviluppo, eseguiamo anche test manuali sulle nuove funzionalità e correzioni di bug prima che vengano distribuite sui nostri server di produzione. Questo ci permette di individuare eventuali regressioni che potrebbero essere state introdotte dal nuovo codice.

L. Separazione degli ambienti

Abbiamo completamente separato i nostri sistemi di sviluppo e di produzione per garantire la sicurezza e per evitare che eventuali problemi di sicurezza possano influire sui nostri sistemi di produzione durante il ciclo di vita dello sviluppo.

Il nostro sistema di sviluppo viene utilizzato per codificare e testare nuove funzionalità o modifiche. Una volta sicuri che una modifica sia pronta, viene mandata al nostro sistema di produzione. Questa separazione garantisce che i nostri utenti non riscontrino mai discontinuità di servizio e che tutte le nuove funzionalità o modifiche siano attentamente testate prima di essere rese disponibili a tutti.

6. Gestione dell'Accesso e dell'Autenticazione

6.1 Accesso alle segnalazioni nella piattaforma:

L'accesso alle segnalazioni viene concesso caso per caso. Ciò significa che due diverse segnalazioni, anche se appartenenti alla stessa categoria e allo stesso reparto, potranno avere utenti diversi che potranno accedervi.

Immaginate la seguente configurazione:

Dipartimenti:

Dipartimento	Utenti
Danimarca	Magnus, Jakob
Svezia	Magnus, Pietro

Categorie:

Categoria	Utenti che riceveranno i nuovi rapporti
Bullismo	Magnus, Jakob, Peter
Corruzione	Magnus

L'accesso iniziale al caso è costituito dagli utenti impostati come destinatari delle categorie e del reparto:

Caso	Dipartimento	Categoria	Accesso iniziale
Caso A	Danimarca	Bullismo	Magnus, Jakob
Caso B	Danimarca	Corruzione	Magnus
Caso C	Dipartimento, Svezia	Bullismo	Magnus, Jakob, Peter

L'accesso dopo la creazione della segnalazione può sempre essere modificato in modo che, ad esempio, per il caso B, Magnus possa dare a Jakob le autorizzazioni per visualizzarlo.

6.2 Provisioning degli utenti e le loro autorizzazioni

Il provisioning degli utenti può essere effettuato solo all'interno del prodotto. Le autorizzazioni degli utenti possono essere gestite in base all'utente:

- Accesso ai reparti e agli utenti (creazione di nuovi utenti, ecc.)
- Accesso alle occorrenze
- Accesso alle statistiche
- Accesso alle impostazioni
- Accesso alle impostazioni di sicurezza

6.3 Fornitori di login disponibili

Al momento sono disponibili diversi fornitori di login che possono essere attivati e disattivati a seconda delle necessità.

Durante l'autenticazione non è disponibile il provisioning degli utenti. Prima di poter accedere, utilizzando un fornitore di login, gli utenti devono essere creati nella piattaforma Whistleblower Software.

6.3.1 Email e password

Il metodo di autenticazione predefinito è l'utilizzo di e-mail e password. L'autenticazione a due fattori può essere inoltre attivata per il singolo utente o applicata a tutti gli utenti del sistema.

Le password vengono memorizzate in formato hash utilizzando bcrypt con 12 round.

L'autenticazione con e-mail e password può essere disattivata se al suo posto si utilizza OAuth 2.0 o SAML 2.0.

Sono in uso i seguenti requisiti per le password:

- È richiesto un punteggio minimo di 70

La funzione di punteggio:

Il punteggio viene calcolato in base ai seguenti elementi:

- Se e quante lettere uniche vengono utilizzate
- Se e quante lettere maiuscole e minuscole, diverse l'una dall'altra, vengono utilizzate
- Se e quanti caratteri speciali vengono utilizzati

La password deve contenere un numero minimo di 8 caratteri (con lettera speciale, numero superiore e lettere minuscole).

6.3.2 OAuth 2.0

Per l'autenticazione OAuth 2.0 sono supportati i seguenti provider:

- Google Identity
- Piattaforma di identità Microsoft (Azure Active Directory)

6.3.3 SAML 2.0

È possibile "installare" il proprio identity provider SAML 2.0 direttamente dal prodotto. Trovate le guide all'installazione per:

- AWS
- Google
- Microsoft
- Ping Identity

Nella guida alla configurazione di SAML 2.0 disponibile [qui](#).

7. Gestione delle vulnerabilità

Whistleblower Software prende sul serio vulnerabilità di qualsiasi natura. Analisi approfondite e misure estese sono messe in atto per evitarle in tutta la nostra infrastruttura, migliorando continuamente il nostro setup.

7.1 Evitare le vulnerabilità

Adottiamo misure per evitare vulnerabilità di alcun tipo ed ogni loro possibile misuse.

Di seguito una panoramica di come evitiamo vulnerabilità e di come ci attiviamo se queste dovessero verificarsi:

7.1.1 Superficie limitata:

La nostra infrastruttura è realizzata in modo tale che vi sia un solo accesso pubblico alla nostra API. Questa superficie è protetta da un WAF (Web application firewall). Ciò assicura che le vulnerabilità, Cross-site scripting (XSS), Remote code execution (RCE), ecc. siano bloccate prima ancora che raggiungano i nostri server API.

Un altro limite della superficie è il nostro approccio per minimizzare la quantità di dipendenze del sistema operativo e rimanere su un sistema operativo piccolo e sicuro. Stiamo costruendo su Alpine Linux.

Un altro punto di sicurezza è che i nostri server web sono container di sola lettura (ad eccezione della directory /tmp) che vengono completamente reimpostati in un nuovo container di istanza rinnovato ogni 15 minuti o durante le distribuzioni. Questo limita la capacità di possibili malware di persistere in generale, ma anche tra le patch e le distribuzioni.

Oltre alla nostra limitata superficie di attacco, utilizziamo per lo più servizi gestiti che assicurano che solo una quantità limitata di configurazioni possa andare storta, assicurando contemporaneamente l'applicazione automatica delle patch.

** L'accesso di Whistleblower Software al database è limitato da un servizio gestito che consente l'accesso solo agli utenti approvati che provengono dalla nostra rete e agli utenti autenticati tramite 2FA.*

7.2 Scansione delle vulnerabilità

Eseguiamo la scansione dell'intero setup, dalle dipendenze open source all'infrastruttura del server, su base giornaliera e in tempo reale. Per la scansione della nostra infrastruttura utilizziamo diversi strumenti approvati e riconosciuti.

Di seguito una panoramica della frequenza e del tipo di ricerca eseguito:

I. Scansioni di vulnerabilità delle dipendenze del codice

Frequenza: Giornaliera

Esaminiamo quotidianamente le nostre applicazioni alla ricerca di vulnerabilità in tutte le loro dipendenze di codice. Lo facciamo utilizzando 4 strumenti diversi, per garantire il risultato più affidabile.

Database utilizzati per la scansione:

- CVE pubblici
- Database privato di vulnerabilità (441% più vulnerabile rispetto al database pubblico successivo più grande).

Esempi di vulnerabilità:

- Dipendenza dall'open source con Cross-site scripting (XSS)
 - Ad esempio: angular <1.8.0
- Dipendenza dall'open source con Esecuzione di codice remoto (RCE)

II. Scansione della vulnerabilità del contenitore/server

Frequenza: Continua e on push

Esaminiamo i nostri contenitori alla ricerca di vulnerabilità nei pacchetti del sistema operativo (i pochi che abbiamo, dato che ci basiamo su quelli di Alpine Linux).

Database utilizzati per la scansione:

- CVE pubblici
- Database privato di vulnerabilità (441% più vulnerabile rispetto al database pubblico successivo più grande)

Esempi di vulnerabilità:

- Pacchetto OS con buffer overflow
 - Ad esempio OpenSSL <1.1.1l-r0
- Pacchetto OS con esecuzione di codice remoto (RCE)

III. Scansione di vulnerabilità dell'infrastruttura/rete

Frequenza: Almeno ogni 18 ore

La nostra infrastruttura ed il suo setup vengono analizzati alla ricerca di vulnerabilità / configurazioni errate. La nostra infrastruttura viene scansionata rispettando i seguenti standard, creati appositamente per il nostro hosting provider:

- Un sottoinsieme dei requisiti del CIS (Center for Internet Security).
- Un sottoinsieme del Payment Card Industry Data Security Standard (PCI DSS) v3.2.1.
- Foundational Security Best Practices per il provider di hosting.

Esempi di vulnerabilità/errori di configurazione:

- File system pubblico accessibile.
- Crittografia mancante in transito / a riposo.
 - Ad esempio, una coda senza crittografia in transito abilitata.
- 2FA mancante
 - Ad esempio, manca la 2FA per gli utenti con accesso al provider di hosting.
- Accesso non limitato a SSH
 - Un gruppo di sicurezza che consente l'ingresso da 0.0.0.0/0 alla porta 22.

8. Compliance, certificazione e audit

8.1 ISAE 3000 Tipo 2

Relazione ISAE 3000 Tipo 2 del revisore indipendente sulle misure di sicurezza delle informazioni e di protezione dei dati in relazione all'accordo con i responsabili del trattamento dei dati. L'audit ISAE 3000 viene effettuato annualmente.

8.2 ISO 27001

Il certificato ISO27001 e il SoA (Statement of Applicability) sono disponibili su richiesta.

8.3 Test di penetrazione

Ogni anno siamo sottoposti ad un test di penetrazione esterno. Il rapporto del test di penetrazione più recente è disponibile su richiesta.

9. Caratteristiche di Sicurezza dell'Account

9.1 Periodo di conservazione fisso

Il sistema supporta l'impostazione di un periodo di conservazione fisso per i dati. In questo modo i dati vengono eliminati dopo un determinato periodo di tempo. Questa funzione è particolarmente utile per i dati dei casi che devono essere eliminati dopo un certo numero di settimane o di anni dalla loro chiusura.

9.2 Whitelist degli indirizzi IP per il login dell'amministratore

Il sistema supporta l'impostazione di un elenco di indirizzi IP da cui gli amministratori possono accedere al backend. Si tratta di un'importante misura di sicurezza per proteggere da accessi non autorizzati. Una volta che un indirizzo IP è stato inserito nella whitelist, gli amministratori potranno accedere da quell'indirizzo IP.

9.3 Autenticazione a più fattori

Il sistema supporta l'impostazione di autenticazione a più fattori per migliorare la sicurezza.

Questo aggiunge un ulteriore livello di sicurezza, richiedendo agli utenti di fornire non solo una password, ma anche un codice ricevuto via SMS. In questo modo è molto più difficile per malintenzionati accedere agli account, anche quando una password venisse rubata. L'autenticazione a più fattori è una misura di sicurezza importante, che consigliamo a tutti gli utenti di impostare.

È inoltre possibile rendere obbligatorio per tutti gli utenti del sistema l'impostazione dell'autenticazione a più fattori al primo accesso al sistema.

9.4 Liste di controllo degli accessi (ACL)

Il sistema ha una configurazione rigorosa per il controllo degli accessi, in modo da garantire che le segnalazioni siano accessibili solo ai gestori dei casi selezionati. Gli operatori che hanno accesso alla segnalazione possono scegliere di condividere il caso con altri operatori, se necessario.

Ogni segnalazione nella piattaforma ha una propria lista di controllo degli accessi.

9.5 Single Sign-On (SSO)

Il sistema supporta un'ampia gamma di opzioni di login SSO. Attualmente sono supportati due protocolli standard del settore: OAuth 2.0 e SAML 2.0. Quando si abilita un'opzione SSO per l'accesso, è possibile disabilitare completamente le funzionalità di reimpostazione della password e di accesso con password nel nostro sistema per gli utenti.

9.5.1 OAuth 2.0

Il protocollo OAuth 2.0 per l'autenticazione è supportato per Microsoft AD e Google. Nella pagina di accesso, è visibile un pulsante "Accedi con Google" / "Accedi con Microsoft" che consente l'autenticazione avviata dal fornitore del servizio (SP).

9.5.2 SAML 2.0

Il protocollo SAML 2.0 per l'autenticazione è supportato da quasi tutti gli identity provider. Non forniamo alcun provider di identità (IdP) SAML 2.0 predefinito, poiché è possibile aggiungere e configurare nuovi provider di identità in maniera autonoma. Si noti che attualmente è supportata solo l'autenticazione avviata da IdP. Ciò significa che l'utente deve effettuare l'accesso all'identity provider prima di essere reindirizzato al nostro sistema.

10. Protezione dell'Anonimato

10.1 Nessuna memorizzazione dell'indirizzo IP o dell'ID hardware del segnalante

Il sistema garantisce che non vengano memorizzate informazioni che non siano state fornite esplicitamente dal segnalante. Nella pagina di segnalazione non vengono utilizzati cookies o tracciamenti di alcun tipo, il che significa che il sistema non memorizza l'indirizzo IP o l'ID del dispositivo utilizzato per la segnalazione.

10.2 Rimozione dei metadati

Il sistema può rimuovere automaticamente i metadati da tutti i file caricati nel sistema. I metadati possono contenere informazioni su:

- ora e luogo di creazione del file,
- autore del file,
- attributi di formattazione e molto altro ancora.

Poiché alcuni metadati potrebbero rivelare l'identità dell'informatore, il sistema è in grado di rimuoverli automaticamente prima che il file venga inviato ai responsabili del caso.

10.3 Distorsione della voce

Per migliorare la tutela dell'anonimato quando l'informatore sceglie di riferire verbalmente, il sistema può distorcere automaticamente la voce del segnalante.