

Università Politecnica delle Marche
Dipartimento di Ingegneria dell'Informazione
Facoltà di Ingegneria Informatica e dell'Automazione



Sviluppo, Progettazione e Implementazione di un'applicazione di Ticket Service tramite la tecnologia Blockchain

Professore:
Prof. Spalazzi Luca
Dott. Spegni Francesco

Studenti:
Denis Bernovschi
Lorenzo Fratini
Emanuele Incicco
Federico Miscia
Andrea Pinciaroli

ANNO ACCADEMICO 2020/2021

Indice

1	Introduzione al progetto	4
2	Requirement Engineering	5
2.1	Requirement Analysis	5
2.1.1	Early Requirement Analysis	7
2.1.2	Late Requirement Analysis	9
2.1.3	Architectural Design - Strategic Dependency Model	10
2.1.4	Architectural Design - Strategic Rationale Model	11
2.2	Risk Identification	12
2.2.1	Asset Identification	12
2.3	Risk Analysis	18
2.3.1	Asset Value and Exposure Assessment	18
2.3.2	Threat Identification	20
2.4	Risk Decomposition	22
2.4.1	Attack Assessment	22
2.5	Risk Reduction	30
2.5.1	Control Identification	30
2.5.2	Feasibility Assessment	30
2.5.3	Security Requirement Definition	41
3	Blockchain e Smart Contracts	51
3.1	Distributed Ledger Technology	51
3.2	Blockchain	51
3.3	Smart Contracts	52
3.4	Ethereum	53
4	Design	54
4.1	Secure Design	54
5	Tecnologie Utilizzate	56
5.1	Quorum	56
5.2	Solidity	56
5.3	Truffle	57
5.4	Node.js	57
5.5	Express	57
5.6	Web3	57
5.7	MySQL	57
6	Misure di sicurezza implementate	58
6.1	Interfacciamento dell'applicazione con la <i>blockchain</i>	58
6.2	Login alla webapp	59
6.3	Prevenzione per Cross-Site Request Forgery (CSRF)	59
6.4	Funzione di logging	59
7	Guida Web App	60
7.1	Manuale Utente	60

Elenco delle figure

1	Risk Driven Security Requirements Specification	6
2	Early Requirement Analysis - Strategic Dependency Model	7
3	Early Requirement Analysis - Strategic Rationale Model	8
4	Late Requirement Analysis - Strategic Dependency Model	9
5	Architectural Design - Strategic Dependency Model	10
6	Architectural Design - Strategic Rationale Model	11
7	Identificazione degli assets sul diagramma i*	13
8	Asset Identification	14
9	Tabella di Jacobson dell'asset Biglietto	15
10	Tabella di Jacobson dell'asset Evento	15
11	Tabella di Jacobson dell'asset API per verifica/invio pagamento	16
12	Tabella di Jacobson dell'asset API per la generazione del Sigillo	16
13	Tabella di Jacobson dell'asset invalida Biglietto	17
14	Asset Table	18
15	Asset Value Assessment & Exposure Assessment	19
16	Threat Identification	20
17	Abuse Case	21
18	Misuse Case	22
19	Attack Tree Abuse Case	23
20	Clumsy Reseller	24
21	Clumsy Invalidator	24
22	Clumsy Ticket Buyer	24
23	Clumsy Event Manager	25
24	Tabella di Jacobson relativa ad Accesso non autorizzato	25
25	Tabella di Jacobson relativa a Danneggiamento dati	25
26	Attack Assessment pt.1	26
27	Attack Assessment pt.2	27
28	Attack Assessment pt.3	27
29	Attack Assessment pt.4	28
30	Mitigation Table	29
31	Mitigation Table	31
32	Mitigation Table	32
33	Feasibility Assessment pt.1	33
34	Feasibility Assessment pt.2	34
35	Feasibility Assessment pt.3	35
36	Feasibility Assessment pt.4	36
37	Feasibility Assessment pt.5	37
38	Feasibility Assessment pt.6	38
39	Feasibility Assessment pt.7	39
40	Tabella di Jacobson aggiornata relativa ad Accesso non autorizzato	40
41	Tabella di Jacobson aggiornata relativa a Danneggiamento dati	40
42	Value to cost Ratios	42
43	Prioritize Requirements pt.1	43
44	Prioritize Requirements pt.2	44
45	Prioritize Requirements pt.3	45
46	Prioritize Requirements pt.4	46
47	Valutazione Misure di Controllo	48
48	Security Requirements Definition	49
49	Security Requirements Definition - Scelte Effettuate	50
50	Workflow di una blockchain	51
51	Logo Quorum	56
52	Solidity Logo	56
53	Truffle Logo	57
54	Webapp Home page	60

55	Form New User	60
56	User Page	61
57	Navigation Bar	61
58	Error Navigation	62
59	Lista degli Eventi	62
60	Dettagli Evento	62
61	Acquisto Biglietto	63
62	Errore Acquisto Biglietto	63
63	Acquisto Biglietto Corretto	63
64	Lista Biglietti Venduti	64

1 Introduzione al progetto

Lo studio da cui ha avuto origine questo elaborato riguarda la progettazione e la realizzazione di un sistema informatico dedicato alla vendita di biglietti per la partecipazione ad eventi di qualsiasi natura. In particolare, per conseguire l'obiettivo del progetto, si sfrutterà la tecnologia *Blockchain*. La tecnologia Blockchain, su cui sarà basata l'intera struttura software, in fase di progettazione, facilita il lavoro collaborativo supportato da computer abilitando flussi di informazioni trasparenti, autenticando la cronologia delle modifiche delle informazioni. In fase di costruzione tale tecnologia può migliorare l'affidabilità e l'attendibilità degli eventi e dei biglietti per tali eventi. Negli ultimi decenni la tecnologia Blockchain si è posta come alternativa a Databases e sistemi di Cloud Storage e, in seguito, attraverso l'utilizzo degli Smart Contracts, ai contratti tradizionali ed ai software tradizionali permettendo la realizzazione delle *Decentralized Applications* (DAPPs). Ad ogni modo, nel corso dell'elaborato viene giustificata tale scelta, esprimendone i vantaggi.

2 Requirement Engineering

2.1 Requirement Analysis

Requisiti generali Elemento centrale, a cui va prestata particolare attenzione, è l'evento, questo memorizza i dati relativi ad un determinato evento compresa la lista dei biglietti venduti per lo specifico evento.

Le diverse azioni che è possibile compiere attraverso il nostro sistema, in sintesi, sono: la creazione di un evento, che fa capo all'Event Manager, la gestione del "Ticket Office" che prevede come sotto-azioni la vendita dei biglietti e la verifica dei pagamenti, il Ticket Invalidator che ha il compito di invalidare il biglietto degli utenti all'ingresso dell'evento ed infine non possiamo non considerare il Ticket Buyer il quale attraverso il nostro sistema può acquistare biglietti per partecipare agli eventi. Ne consegue logicamente che il biglietto, risorsa chiave del sistema, deve essere soggetto a verifiche di autenticità. Per garantire quanto detto precedentemente è stato previsto l'utilizzo di moduli esterni che lavorano off-chain, in particolare per la generazione del sigillo e la verifica del pagamento; tali processi si svolgono attraverso delle opportune API. Per quanto concerne la generazione del sigillo, abbiamo deciso di affidarci ad una API esterna, la quale in seguito alla verifica del pagamento effettuerà la generazione del sigillo. Per quanto riguarda la verifica del pagamento l'API esterna restituirà true o false simulando un pagamento che, rispettivamente, è andato a buon fine o meno, in quanto in questo progetto non si è preso in considerazione il pagamento che gli utenti effettuano per acquistare i biglietti degli eventi. Per modellare i requisiti richiesti è stato inizialmente utilizzato il linguaggio di modellazione **I***.

Linguaggio di Modellazione I* I* (i-star) è un linguaggio di modellazione dei requisiti più ricco rispetto ai meglio conosciuti **casi d'uso di UML** che non sono in grado di esprimere al completo tutte le relazioni tra le componenti del progetto da realizzare. Gli elementi che caratterizzano il *linguaggio i** sono i seguenti:

- **ATTORE**: viene generalmente indicato con un tondo ed è usato per rappresentare uno stakeholder interno o esterno. Si può specializzare l'attore andandone ad esprimere l'agente (individuo specifico) e il ruolo (figura ricoperta nel contesto di modellazione). Per ogni attore, in I* è necessario andare a definire quelle che sono le loro attitudini, evidenziate nel seguito.
- **SOFTGOAL**: è indicato con un simbolo a forma di nuvola e rappresenta un obiettivo per cui non vi è un criterio preciso che permetta di stabilire se effettivamente sia stato raggiunto o meno, in virtù del fatto che l'obiettivo viene espresso in forma piuttosto vaga.
- **HARDGOAL**: ha un ovale come simbolo e rappresenta sempre un obiettivo, questa volta più preciso, per cui si è in grado di stabilire dei criteri di raggiungimento.
- **TASK**: simboleggiato con un esagono indica una particolare attività eseguibile da un determinato attore per raggiungere uno specifico obiettivo.
- **RISORSE**: vengono indicate con dei rettangoli e fanno riferimento ad elementi prodotti dalla realizzazione di un task o risorse necessarie al fine di poter eseguire un'attività.

In i* è fondamentale esprimere quelle che sono le **dipendenze** tra le varie componenti. Queste vengono espresse attraverso degli archi contrassegnati con una D che evidenziano proprio il verso del flusso di dipendenza (e non il verso del flusso delle informazioni che magari potrebbe essere inverso alla dipendenza). Le dipendenze possono legare tra di loro solo attori attraverso quattro tipologie di dipendenza: *goal dependency*, *task dependency*, *resource dependency* e *softgoal dependency*, ciascuna delle quali indica qual è l'elemento (*dependum*) che va a legare la dipendenza tra due attori (uno dei quali è il *depender* e l'altro il *dependee*).

La forza del linguaggio I* è che permette di eseguire delle decomposizioni in modo tale da avere un diagramma estremamente dettagliato in cui è possibile mettere in mostra tutte le relazioni di dipendenza. Le principali forme di decomposizione sono:

- **GOAL DECOMPOSITION**: si decompone l'obiettivo in tanti sotto-obiettivi che possono essere legati da una relazione di and (and decomposition) in cui devono essere raggiunti tutti i sotto-obiettivi per concorrere al raggiungimento dell'obiettivo sovrastante, oppure da una relazione di or (or decomposition) in cui basta raggiungere almeno uno dei sotto-obiettivi.

2.1 Requirement Analysis

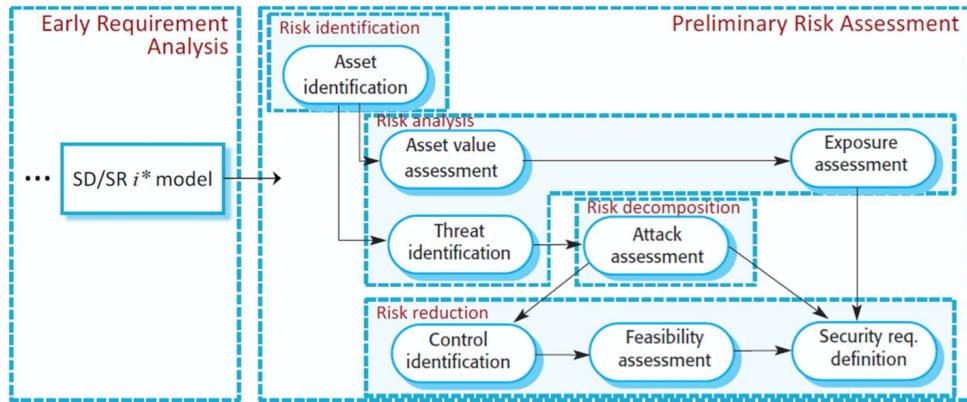


Figura 1: Risk Driven Security Requirements Specification

- MEANS-END DECOMPOSITION: una determinata attività è un mezzo per raggiungere un determinato fine. Le attività sono da intendersi congiuntive, mentre se si vuole esprimere un legame disgiuntivo è necessario passare per i sotto-obiettivi visti precedentemente prima di usare questa tipologia di decomposizione.
- SOFTGOAL DECOMPOSITION: si va ad esprimere il modo in cui vari elementi (obiettivi, risorse o altre attività) aiutano nel perseguitamento di quell’obiettivo vago, esprimendolo attraverso dei simboli (+ o -) o attraverso delle parole chiave. Ad esempio, la parola "help" (o, in modo equivalente,++) indica che quell’elemento va nella direzione del raggiungimento del softgoal.
- TASK DECOMPOSITION: indica tutto ciò che deve essere effettuato prima della realizzazione di una particolare attività, cioè per poter compiere un task è necessario prima acquisire una determinata risorsa o aver raggiunto uno specifico obiettivo.

*i** permette inoltre di andare a definire qual è l’ambito di competenza di un determinato attore, andando a identificare nello Strategic Rationale Modeling il boundary associato a quell’attore. Questo particolare modello, in cui generalmente si trova una decomposizione raffinata di tutte le attitudini dell’attore, unito alle dipendenze dei vari attori del sistema, permette di comprendere tutte quelle che sono le dinamiche di interazione all’interno del sistema. Tutta la parte progettuale relativa alla raccolta, all’analisi e alla modellazione dei requisiti è stata, infatti, eseguita attraverso l’uso di questo linguaggio di modellazione come supporto.

Risk Driven Security Requirements Specification Nella Fig.1, possiamo osservare lo schema Risk Driven Security Requirements Specification, in particolare sono rappresentate le fasi più rilevanti da dover svolgere al fine di ottenere una corretta progettazione di un software sicuro. Successivamente ciascuna fase verà affrontata nel dettaglio con tutte le procedure necessarie al suo completamento.

2.1 Requirement Analysis

2.1.1 Early Requirement Analysis

Per poter comprendere meglio quali sono gli attori e i legami che agiscono in tale ambito inizialmente si ricorre alla Early Requirement Analysis, letteralmente Analisi Preliminare dei Requisiti. In questa fase si analizza il caso reale, o meglio l'aspetto sociale, e si vede ciò che ci si aspetta da ciascun attore. IN questo caso risulta di fondamentale importanza non considerare il sistema software che verrà poi implementato, l'obiettivo di questa fase infatti è quello di comprendere il funzionamento in assenza del sistema software. Possiamo osservare il diagramma nella Figura Fig. 2

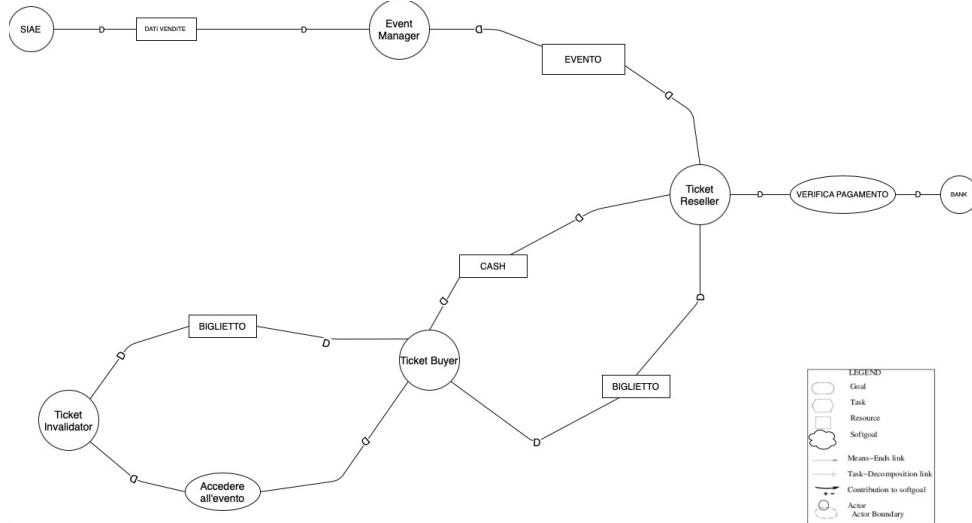


Figura 2: Early Requirement Analysis - Strategic Dependency Model

Come anticipato nelle sezioni precedenti, ogni diagramma è stato realizzato seguendo i criteri tipici del linguaggio di modellazione i^* . La figura su riportata indica il modello i^* dell'Early Requirements Analysis in cui si unisce lo strategic rationale model al flusso delle dipendenze. Dal diagramma precedente emergono i quattro attori principali: Event Manager, Ticket Reseller, Ticket Buyer e Ticket Invalidator, oltre a due attori secondari: Bank e Siae, questi due attori non verranno considerati successivamente in quanto non di importanza principale ai fini del progetto.

- **Event Manager:** Il suo compito è la gestione degli eventi. Si relaziona soltanto con il Ticket Reseller.
- **Ticket Reseller:** La sua funzione è la vendita dei biglietti, per svolgerlo si interfaccia con il Ticket Buyer e con l'Event Manager, quest'ultimo gli fornirà le informazioni sull'evento.
- **Ticket Buyer:** L'utente che intende partecipare all'evento ed acquista biglietti. Si relaziona con il Ticket Reseller e il Ticket Invalidator.
- **Ticket Invalidator:** Si occupa di invalidare il biglietto del Ticket Buyer all'ingresso dell'evento. Si relaziona soltanto con il Ticket Buyer.

Di seguito (Fig.3) si riporta il diagramma, ottenuto dalla Early Requirement Analysis, comprensivo dei Boundary dei vari attori.

2.1 Requirement Analysis

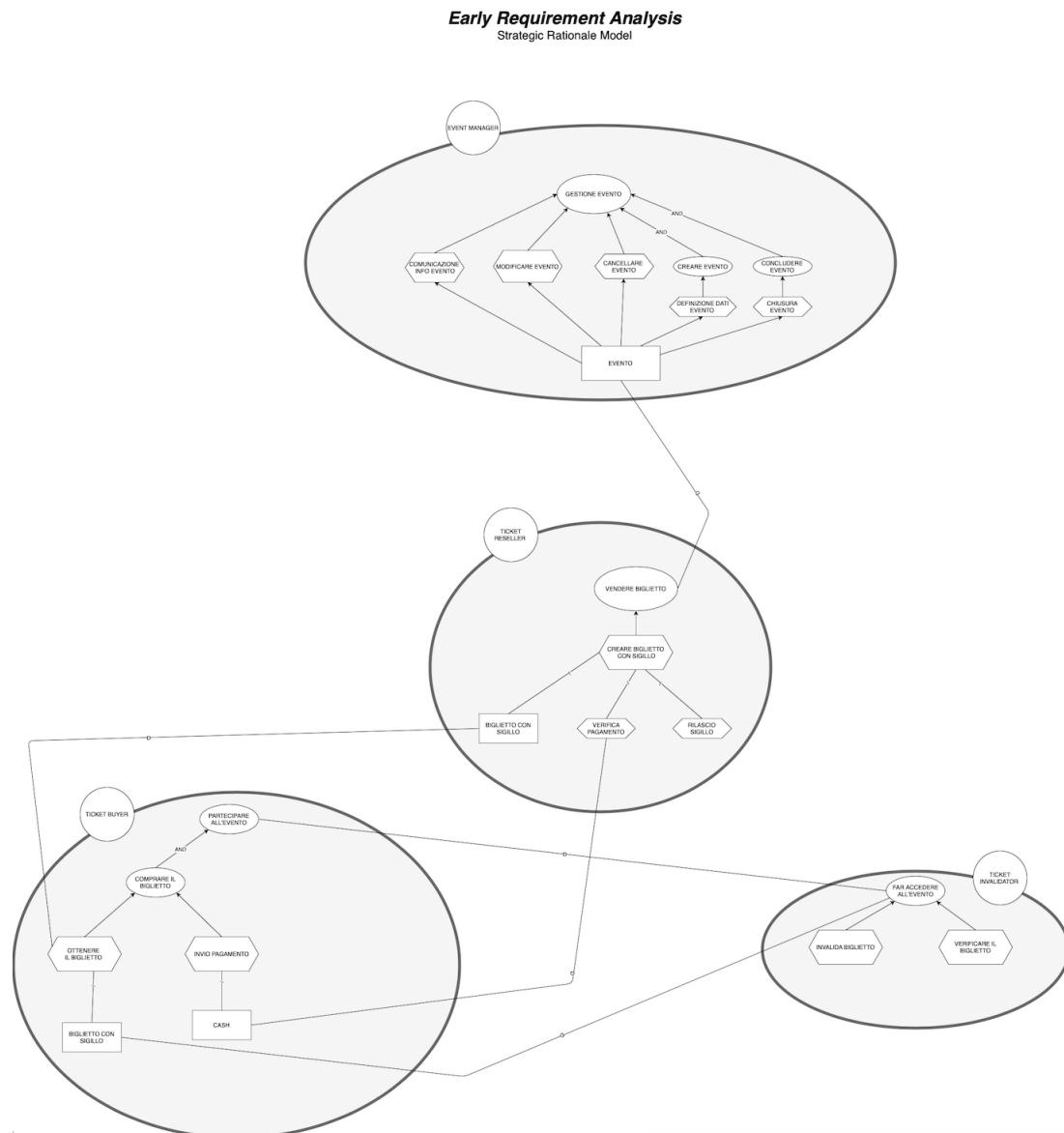


Figura 3: Early Requirement Analysis - Strategic Rationale Model

2.1 Requirement Analysis

2.1.2 Late Requirement Analysis

Letteralmente Analisi dei Requisiti Successiva, si basa sul prendere in considerazione l'introduzione di un sistema software che verrà considerato come un attore all'interno del modello. Così facendo si genereranno nuovi flussi di dipendenza dal momento che si ragiona sulle attività che dovranno essere delegate al software stesso. Tutto ciò che sarà inserito all'interno del boundary del sistema software, diventa un requisito funzionale. Il diagramma è riportato di seguito, Fig.4.

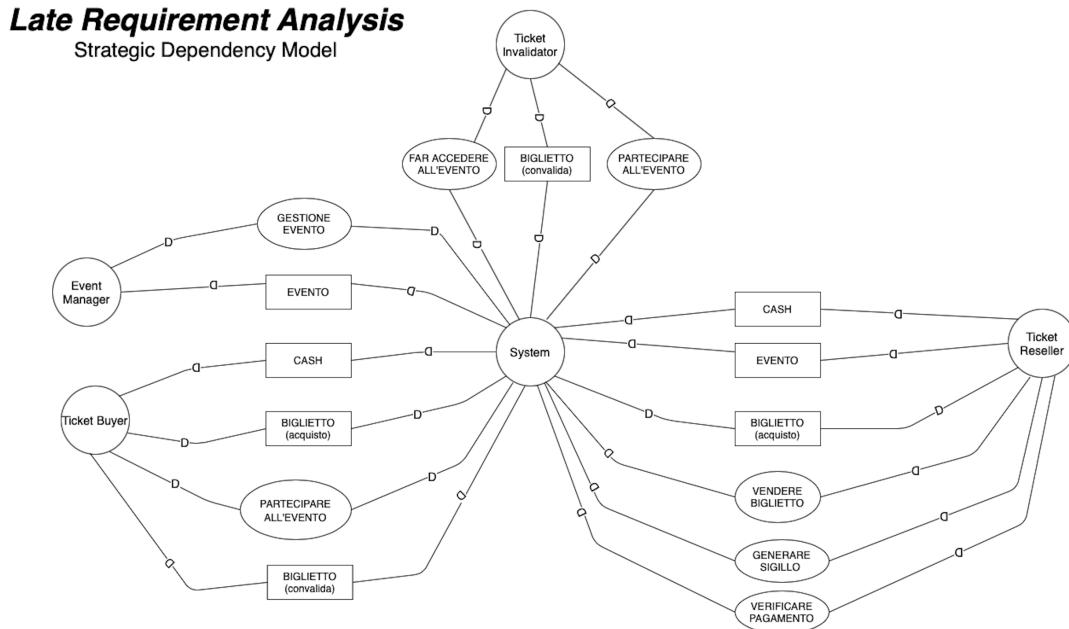


Figura 4: Late Requirement Analysis - Strategic Dependency Model

Il sistema in questione crea un evento attraverso l'attore *Event Manager*. Successivamente l'attore *Ticket Reseller* andrà a leggere e scrivere sull'evento le varie operazioni come la vendita biglietti e la generazione sigillo. Il *Ticket Buyer* dovrà interagire con il sistema per l'acquisto del biglietto. Mentre il *Ticket Invalidator* interagisce con il sistema al momento dell'invalidazione dei biglietti, ovvero quando il *Ticket Buyer* accede all'evento. Per quanto riguarda il processo di pagamento e di generazione del sigillo, è il sistema che verifica il pagamento e genera il sigillo attendendo l'esito.

2.1 Requirement Analysis

2.1.3 Architectural Design - Strategic Dependency Model

In questa sezione sarà effettuata una prima suddivisione del sistema in moduli per capire come le sue componenti interagiscono fra loro, costituendo l'Architectural Design. L'Architectural Design permette di focalizzarsi sui singoli componenti del sistema, andando ad individuare il flusso operativo e tutte le possibili vulnerabilità sfruttabili da un attaccante.

Ecco quindi che, un intero software può essere visto come un insieme di più componenti che interagiscono fra loro. In base al numero di componenti e alle relative interazioni si definisce l'architettura. L'attore *System* inizia così a delegare compiti ad altri attori interni che diventano quindi dei moduli dello stesso. La delega avviene sulla base degli obiettivi che devono essere perseguiti dal sistema. Una rappresentazione della divisione del sistema in moduli è visibile nella Fig.5.

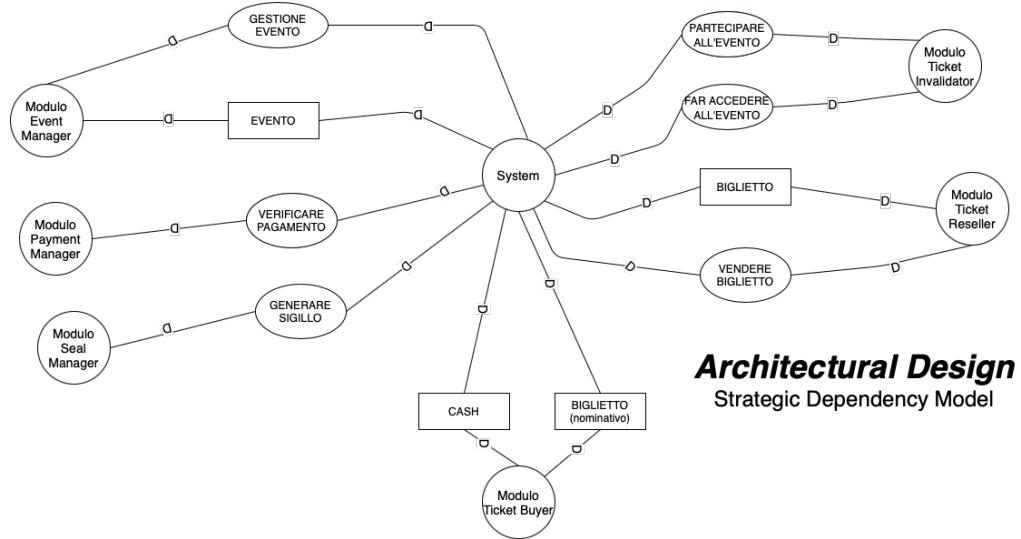


Figura 5: Architectural Design - Strategic Dependency Model

Dopo aver definito i moduli in cui il sistema software si divide si sono analizzati ciascuno di loro, andando a vedere le interazioni che questi hanno con il sistema.

- Modulo Event Manager - Si occupa della gestione dell'evento e opera sulla risorsa EVENTO.
- Modulo Payment Manager - Incaricato di verificare il pagamento.
- Modulo Seal Manager - Incaricato di generare il sigillo.
- Modulo Ticket Buyer - Si occupa di effettuare il pagamento (risorsa CASH¹) e si aspetta la risorsa BIGLIETTO².
- Modulo Ticket Reseller - Si occupa di vendere il biglietto, scambia la risorsa BIGLIETTO.
- Modulo Ticket Invalidator - Si occupa di invalidare il biglietto al momento dell'utilizzo, cioè all'ingresso dell'evento da parte del Ticket Buyer.

¹Per CASH si intende il denaro in generale e non soltanto il denaro in contanti.

²Il biglietto è comprensivo di un identificativo dell'acquirente e del sigillo.

2.1 Requirement Analysis

2.1.4 Architectural Design - Strategic Rationale Model

In questa sezione si è deciso di rappresentare lo *Strategic Rationale Model*, cioè di rappresentare più nel dettaglio i ruoli di ciascuno dei moduli precedentemente individuati. Ciò è stato realizzato andando a rappresentare i *boundary* di ogni modulo del sistema software.

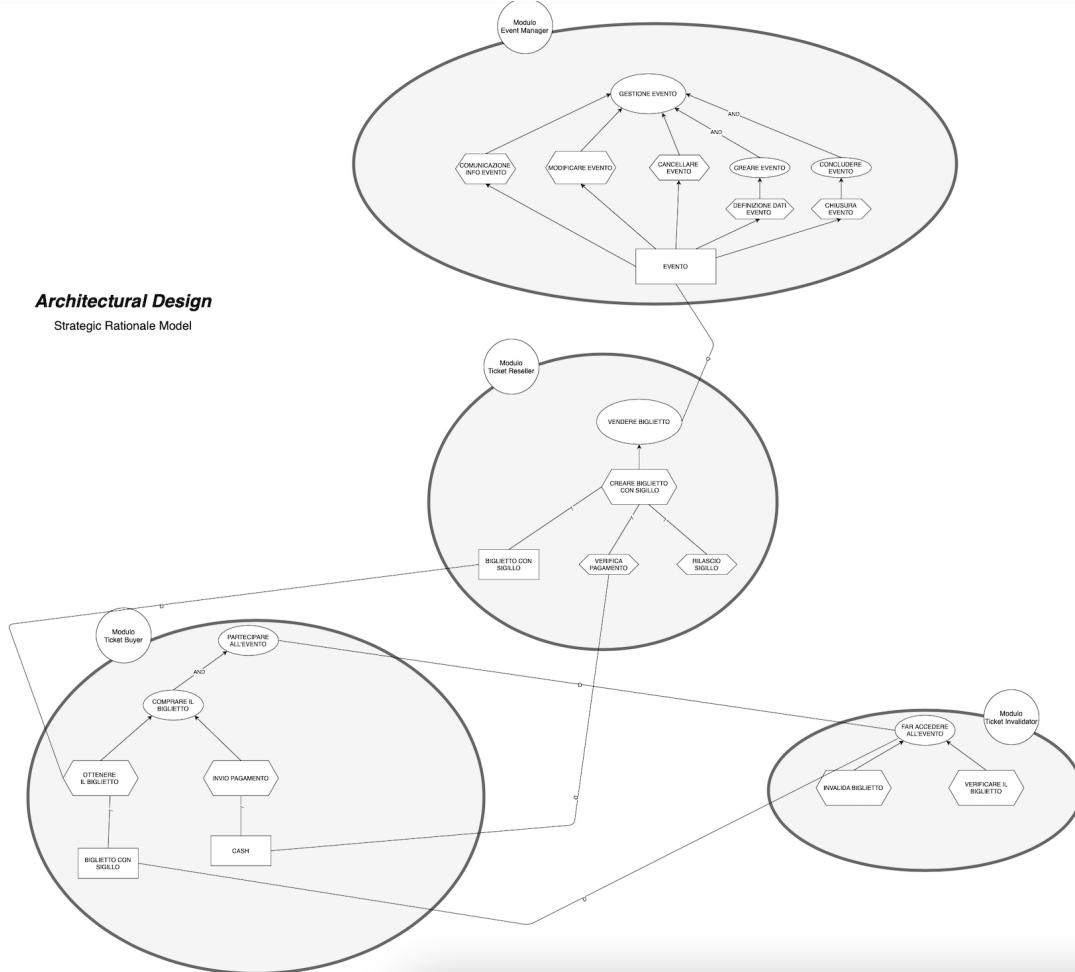


Figura 6: Architectural Design - Strategic Rationale Model

2.2 Risk Identification

2.2 Risk Identification

In questa fase si analizzano le policy di sicurezza, ovvero l'insieme di quelle regole che governano tutti gli aspetti legati alla sicurezza. Per prima cosa è necessario identificare gli asset, cioè tutto ciò che ha importanza all'interno dell'organizzazione e che quindi merita di essere protetto da eventuali attacchi di tipo cyber o minacce di vario genere. Vanno poi stabiliti per ogni asset gli obiettivi di sicurezza che si vogliono perseguire ed infine quali sono le politiche a livello organizzativo, cioè comprendere quali sono le operazioni che i vari utenti possono o non possono svolgere su quel determinato asset. Nella figura 8 viene riportata la tabella associata all'identificazione degli assets.³

Gli assets possono essere tangibili e non, si hanno: Biglietto ed Evento che sono assets tangibili, elementi cruciali per il sistema. Gli altri assets sono non tangibili, in quanto beni immateriali ma comunque fondamentali, in quanto sono cruciali per il corretto funzionamento del sistema.

Per quanto riguarda gli obiettivi delle policy di sicurezza sono state utilizzate le triadi conosciute, ovvero:

CIA TRIADE

- Confidenzialità: una certa informazione non deve essere accessibile ad utenti non autorizzati;
- Integrità: una certa informazione non deve essere modificata da utenti non autorizzati;
- Disponibilità: una certa informazione deve essere sempre disponibile quando richiesta da utenti autorizzati.

AAA TRIADE

- Autenticità: una certa informazione è autentica e vera;
- Garanzia: gli utenti si comportano come desiderato;
- Responsabilità: è sempre possibile attribuire la responsabilità di una determinata azione compiuta all'interno del sistema al soggetto che l'ha realizzata.

TRIADE SAFETY/RELIABILITY/RESILIENCE

- Safety: il sistema non reca danno a cose e persone;
- Affidabilità: il sistema eroga un servizio come gli utenti si aspettano;
- Resilienza: garantire una continuità di servizio anche in seguito ad errori e guasti.

2.2.1 Asset Identification

Nella Fig.7 sono evidenziati gli assets individuati, gli assets sono stati selezionati sulla base delle principali risorse e/o attività che sono critiche per il nostro applicativo. Ecco quindi che gli assets sono i seguenti:

1. Gestione Evento;
2. Evento;
3. Creare Biglietto (con sigillo);
4. Biglietto (con sigillo);
5. Verifica/Invio Pagamento;
6. Rilasciare Sigillo;
7. Ottenere Biglietto;
8. Invalida Biglietto.

2.2 Risk Identification

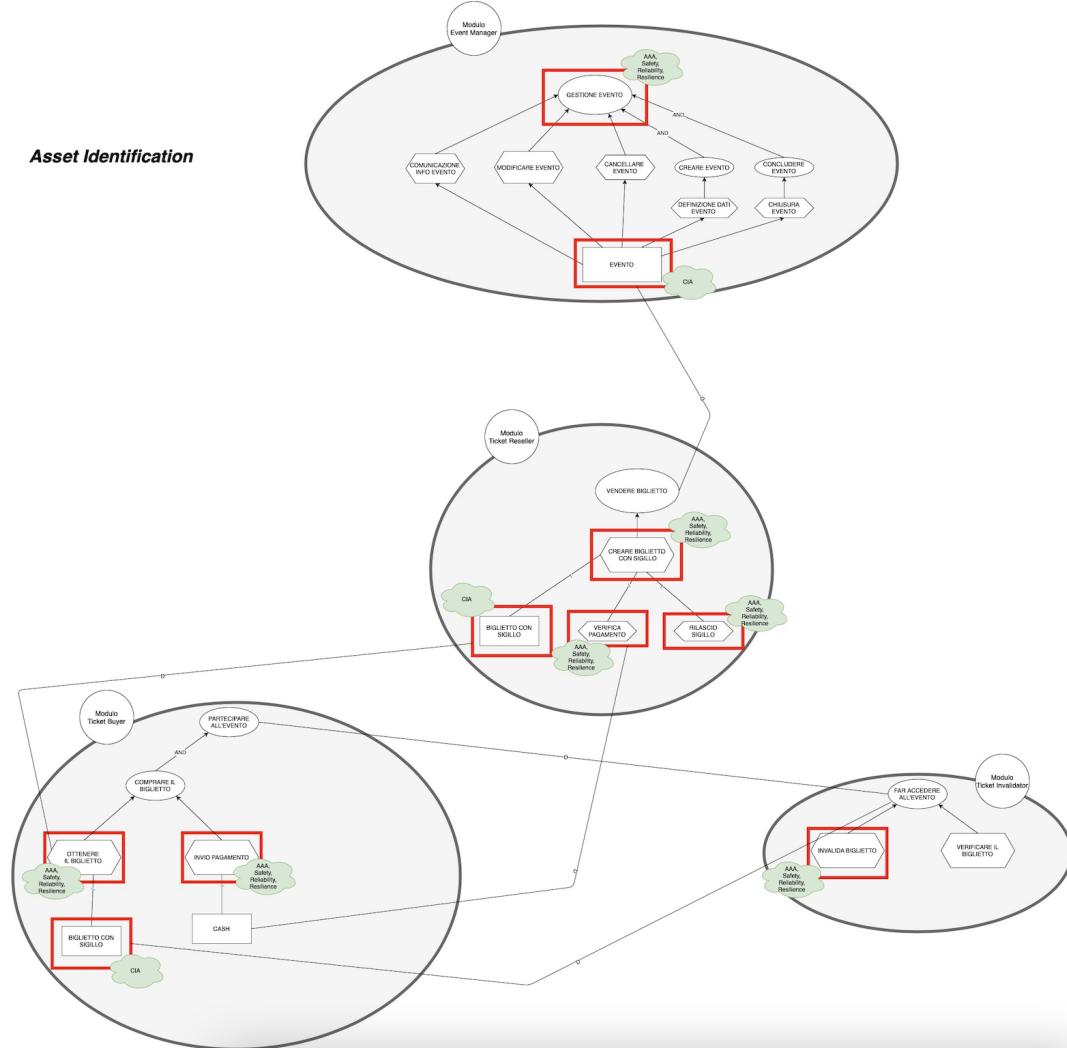


Figura 7: Identificazione degli assets sul diagramma i*

Per ciascuno di questi assets siamo andati a definire gli obiettivi delle security policy da rispettare, riportando anche quest'ultime nella Fig.7 accanto all'assets nel softgoal verde.

Successivamente, nella tabella Tab.8, sono riportate, oltre agli obiettivi di security policy da rispettare, anche le policy a livello organizzativo per ciascun asset.

³L'invio e la verifica del pagamento non sono stati implementati in quanto non importanti ai fini del progetto, si utilizzerà un'API apposita che simula un pagamento che è andato a buon fine o meno.

2.2 Risk Identification

ASSET IDENTIFICATION		POLICY DI SICUREZZA	POLICY ORGANIZZATIVE
BIGLIETTO		Confidenzialità Integrità Disponibilità	Il Ticket Buyer acquista il biglietto ed ottiene la risorsa BIGLIETTO in seguito all'esito della verifica del pagamento. Il ticket reseller si incarica quindi della verifica del pagamento tramite l'API "Verifica del pagamento" e avvia la procedura di generazione del sigillo tramite l'API relativa ad essa. L'utente infine potrà soltanto ricevere il biglietto comprensivo di sigillo, senza interferire in alcun modo con il giornale dei eventi.
EVENTO		Confidenzialità Integrità Disponibilità	Il Ticket Reseller può soltanto leggere le informazioni sull'evento. Tutti gli altri attori (a parte l'Event Manager) non possono neanche leggere le informazioni sull'evento.
VERIFICARE/INVIARE PAGAMENTO (API)		Autenticità Garanzia Responsabilità Incolmunità (Safety) Affidabilità Resilienza	L'API "Verifica del pagamento" viene utilizzata dal ticket reseller (unico utilizzatore in sola lettura) per verificare il pagamento effettuato dal ticket buyer dopo l'acquisto del biglietto. L'API accetta pagamenti elettronici solo tramite circuiti verificati. Deve essere garantita la privacy dei dati del cliente e del pagamento effettuato. La sicurezza minima dei pagamenti è lo standard PS2. Al termine della verifica del pagamento il ticket reseller può procedere alla generazione del sigillo per la risorsa BIGLIETTO e il titolare viene inserito automaticamente nella lista dei registrati all'evento.
RILASCIARE SIGILLO (API)		Autenticità Garanzia Responsabilità Incolmunità (Safety) Affidabilità Resilienza	L'API per il rilascio del sigillo viene invocata dal ticket reseller a seguito della ricezione della notifica di corretto pagamento pertanto egli è l'unico attore a potervi accedere, e comunque solo in lettura e non in scrittura. L'API in questione deve dare inoltre la possibilità al reseller di apporre il sigillo sul biglietto che verrà consegnato al buyer così da ottenere il "non fungible token" (NFT) ma non deve in alcun modo concedere l'opportunità di modificare i dati del biglietto.
INVALIDARE BIGLIETTO		Autenticità Garanzia Responsabilità Incolmunità (Safety) Affidabilità Resilienza	Il Ticket Invalidator presente all'evento, si occuperà tramite uno scanner di verificare che i Ticket Buyer presenti il biglietto comprensivo di Sigillo autentico e di contrassegnarlo come "partecipante all'evento" ai fini di evitare che due Ticket Buyer distinti entriano con il medesimo biglietto
GESTIRE EVENTO		Autenticità Garanzia Responsabilità Incolmunità (Safety) Affidabilità Resilienza	L'Event Manager può eseguire tutte le attività di gestione dell'evento. Tutti gli altri attori non possono gestire l'evento.
CREARE BIGLIETTO		Autenticità Garanzia Responsabilità Incolmunità (Safety) Affidabilità Resilienza	il ticket buyer è colui che ottiene il biglietto, ma la creazione è a carico del ticket reseller
OTTENERE BIGLIETTO		Autenticità Garanzia Responsabilità Incolmunità (Safety) Affidabilità Resilienza	il ticket buyer è colui che ottiene il biglietto, ma la creazione è a carico del ticket reseller

Figura 8: Asset Identification

2.2 Risk Identification

Tabelle di Jacobson per gli Asset In questa sezione riportiamo le tabelle di Jacobson relative ai principali asset individuati precedentemente.

Use case Name:	Biglietto
Actors	Ticket Reseller, Ticket Buyer
Description	Il Ticket Reseller provvede alla verifica del pagamento e successivamente avvia la procedura di generazione del biglietto, infine nell'ultimo passaggio una volta ottenuto il biglietto, invia il tutto al Ticket Buyer
Data	Biglietto
Stimulus and Preconditions	Il biglietto deve essere acquistato Il pagamento deve essere verificato 1. Il Ticket Buyer acquista il biglietto dal Ticket Reseller 2. Il Ticket Reseller verifica il pagamento da parte del Ticket Buyer 3. Il Ticket Reseller avvia la procedura di generazione del Biglietto 4. Il Ticket Reseller avvia la procedura di generazione del sigillo
Basic Flow	
Alternative Flow	1. Il Ticket Buyer acquista il biglietto dal Ticket Reseller 2. Il Ticket Reseller verifica il pagamento e la procedura non va a buon fine 3. L'invocazione della procedura di generazione del sigillo non va
Exception Flow	
Responde and Postconditions	Il Biglietto generato correttamente e si avvia la procedura di generazione sigillo
Non Functional Requirements	Confidenzialità, Integrità, Disponibilità
Comments	Ticket Buyer può solo acquistare il biglietto e ricevere la risposta e/o risorsa, ma non può in nessun modo interferire con il Ticket Reseller, per varie operazioni di scrittura/lettura sul sistema, l'event Manager, si occuperà solo di ricevere il resoconto completo della vendita

Figura 9: Tabella di Jacobson dell'asset Biglietto

Use case Name:	Trasferire informazioni evento
Actors	Event Manager, Ticket Reseller
Description	L'Event Manager invia le informazioni dell'evento al Ticket Reseller.
Data	Evento
Stimulus and Preconditions	L'evento è stato creato e pronto all'invio. Comando emesso da un Event Manager.
Basic Flow	1. L'Event Manager crea l'evento e invia le sue informazioni al Ticket Reseller. 2. Il Ticket Reseller crea e vende i biglietti sulla base delle informazioni ricevute sull'evento.
Alternative Flow	
Exception Flow	1. L'Event Manager crea l'evento e invia le sue informazioni al Ticket Reseller. 2. Le informazioni sull'evento non arrivano al Ticket Reseller.
Responde and Postconditions	I biglietti sono venduti con le informazioni ricevute sull'evento.
Non Functional Requirements	Confidenzialità, Integrità, Disponibilità
Comments	Il Ticket Reseller può soltanto leggere le informazioni sull'evento. Tutti gli altri attori (a parte l'Event Manager) non possono neanche leggere le informazioni sull'evento.

Figura 10: Tabella di Jacobson dell'asset Evento

2.2 Risk Identification

Use case Name:	API Verifica/Invio Pagamento
Actors	Ticket Buyer e Ticket Reseller
Description	Il Ticket buyer invia il pagamento tramite API dopo l'acquisto del biglietto. Il Ticket Reseller utilizza L'API per verificare che il pagamento da parte del ticket buyer sia avvenuto in maniera corretta.
Data	Biglietto
Stimulus and Preconditions	Il ticket buyer deve acquistare il biglietto
Basic Flow	<ol style="list-style-type: none"> 1. Il Ticket Buyer acquista il biglietto dal Ticket Reseller 2. Il ticket buyer invia il pagamento tramite l'API 3. Ticker Reseller utilizza l'API per verificare il pagamento
Alternative Flow	
Exception Flow	<ol style="list-style-type: none"> 1. Il Ticket Buyer acquista il biglietto dal Ticket Reseller 2. Il Ticket buyer utilizza l'API per inviare il pagamento 3. Il Ticker reseller con l'API verifica il pagamento 4. La verifica del pagamento produce un errore(NON VA A BUON FINE)
Responde and Postconditions	L'acquisto del biglietto è confermato e si procede con la generazione del sigillo
Non Functional Requirements	Autenticità, Garanzia, Responsabilità, Incolumità, Affidabilità, Resilienza
Comments	Il corretto funzionamento dell'API Verifica Pagamento può produrre due esiti: Pagamento confermato o Pagamento rifiutato (errore di inserimento dati, insufficiente denaro). L'errore menzionato nell'exception flow corrisponde ad un non corretto funzionamento dell'API dovuto a una situazione di interruzione del servizio di verifica (denial of service).

Figura 11: Tabella di Jacobson dell'asset API per verifica/invio pagamento

Use case Name:	Rilascio del sigillo
Actors	Ticket Reseller, (Ticket Buyer?)
Description	Il Ticket Reseller, a fronte della verifica con esito positivo del pagamento di un biglietto, utilizza l'apposita API per avviare la procedura di generazione del sigillo la quale agisce indipendentemente dal giornale degli eventi. Ad operazione conclusa l'hash contenente il sigillo viene inserito come parte integrante del biglietto da rilasciare all'attore Ticket Buyer.
Data	Biglietto
Stimulus and Preconditions	Il biglietto deve essere corredata di tutte le informazioni sull'evento e sull'acquirente. E' necessario avere una certificazione della corretta transazione per l'acquisto del biglietto
Basic Flow	<ol style="list-style-type: none"> 1. Il Ticket Buyer acquista il biglietto dal Ticket Reseller 2. Il Ticket Reseller verifica il pagamento effettuato dal Ticket Buyer 3. Il Ticket Reseller avvia la procedura di generazione del sigillo fiscale 4. Il Ticket Reseller inserisce il sigillo nel biglietto ed emette quest'ultimo verso il buyer
Alternative Flow	
Exception Flow	<ol style="list-style-type: none"> 1. Il Ticket Buyer acquista il biglietto dal Ticket Reseller 2. Il Ticket Reseller verifica il pagamento 3. La generazione del sigillo non avviene e non è possibile emettere il biglietto completo
Responde and Postconditions	Il biglietto ha prova di autenticità e risulta spendibile all'ingresso dell'evento
Non Functional Requirements	Autenticità, Garanzia, Responsabilità, Incolumità, Affidabilità, Resilienza
Comments	Ticket Buyer, Ticket Reseller e Ticket Validator possono soltanto leggere il sigillo fiscale, ma nessuno di essi lo può modificare/eliminare. L'Event Manager non può fare niente di ciò. La procedura di generazione del sigillo è richiamabile esclusivamente dal Ticket Reseller.

Figura 12: Tabella di Jacobson dell'asset API per la generazione del Sigillo

2.2 Risk Identification

Use case Name:	Invalida Biglietto
Actors	Ticket Buyer, Ticket Validator
Description	Il Ticket Buyer nel momento in cui desidera accedere all'evento, presenta al Ticket Validator il suo biglietto, il quale attua la procedura di invalidazione del biglietto, contrassegnando il biglietto come "utilizzato" evitando così usi impropri
Data	Biglietto
Stimulus and Preconditions	Il ticket buyer deve accedere all'evento
Basic Flow	1. Il Ticket Buyer accede all'evento 2. Il Ticket Validator, verifica il biglietto e lo contrassegna come "utilizzato"
Alternative Flow	
Exception Flow	1. Il Ticket Buyer presenta un biglietto non valido 2. Il Ticket Buyer presenta un biglietto privo di sigillo o con sigillo contraffatto 3. Il Ticket Buyer utilizza un biglietto già contrassegnato come "utilizzato"
Responde and Postconditions	L'ingresso dell'evento procede correttamente e il biglietto viene contrassegnato come "utilizzato"
Non Functional Requirements	Autenticità, Garanzia, Responsabilità, Incolumità, Affidabilità, Resilienza
Comments	Il ticket invalidator, attraverso un scanner verifica che biglietto sia presente nel sistema ed in modo automatico registra l'ingresso all'evento contrassegnando il biglietto come "utilizzato"

Figura 13: Tabella di Jacobson dell'asset invalida Biglietto

2.3 Risk Analysis

2.3 Risk Analysis

2.3.1 Asset Value and Exposure Assessment

Dopo aver identificato gli asset e viste le relative policy da rispettare è possibile andare ad assegnare un valore a ciascun asset e valutare il possibile impatto che ne deriverebbe dal mancato rispetto di ciascun requisito. Per fare ciò è stato scelto di utilizzare una valutazione qualitativa piuttosto che quantitativa, basata su una scala di **Likert** a 3 valori. Questo metodo prevede di esprimere il grado di accordo nei confronti di una determinata affermazione con dei numeri che vanno da 1 a 3, dove 1 esprime il completo disaccordo mentre 3 il totale accordo. Di seguito, nella figura 14, viene riportata la *Asset Table*.

Asset Table				
ASSET	VALORE		IMPATTO	
BIGLIETTO CREARE BIGLIETTO OTTENERE BIGLIETTO	Nevralgici per l'intero sistema. Va prestata particolare attenzione alle diverse sottofasi per ottenere la risorsa biglietto. Potenzialmente critici per la sicurezza.	3	Possibili problematiche comprometterebbero il corretto funzionamento del sistema. Annullamento del biglietto. Possibili danni per il Ticket Buyer.	3
EVENTO	Richiesto per la vendita dei biglietti. Potenzialmente critico per la sicurezza.	3	I biglietti venduti per l'evento in questione devono essere annullati. Costi per il ripristino delle corrette informazioni. Costi per eventuali risarcimenti.	3
VERIFICARE/INVIARE PAGAMENTO (API)	Richiesto per ottenere il biglietto. Potenzialmente critico per la sicurezza.	3	Il biglietto non può essere correttamente acquistato. Costi per ripristinare il sistema.	3
RILASCIARE SIGILLO (API)	Richiesto per la prova di originalità del biglietto. Potenzialmente critico per l'effettivo accesso all'evento	3	Il biglietto non può essere emesso. E' necessario ripetere l'operazione. Possibile danno per il Ticket Buyer	3
INVALIDARE BIGLIETTO	La verifica dell'autenticità del biglietto non pone sotto scacco il sistema da possibili attacchi durante la fase di registrazione all'ingresso del evento	2	Un biglietto non convalidato correttamente apre a due possibili problematiche: un Ticket Buyer non può accedere all'evento nonostante in regola oppure un Ticket Buyer non autorizzato accede all'evento	2
GESTIRE EVENTO	Richiesto per l'organizzazione e la pianificazione degli eventi. Potenzialmente critico.	3	Gli eventi devono essere momentaneamente sospesi (non è possibile acquistare biglietti). Costi per il ripristino. Perdite in termini di vendite.	3

Figura 14: Asset Table

Le policy di sicurezza che erano state individuate in fase di asset identification vengono ora espresse sotto forma di veri e propri requisiti del sistema per i quali è possibile implementare specifiche misure per gestirli. Ciascun asset è stato classificato con un valore. Nella tabella della Fig. 15 troviamo riportati i requisiti del sistema per ciascun asset precedentemente identificato. Associato a ciascun requisito abbiamo un valore, che rappresenta il valore aggiunto nell'implementare lo specifico requisito del sistema. Troviamo poi il valore dell'impatto provocato dalla violazione di ciascuno dei requisiti. Da evidenziare l'elevato impatto che hanno le violazioni di integrità per i biglietti e gli eventi. Va inoltre osservato il valore elevato per la resilienza della gestione evento, una mancata osservazione di questo requisito comporterebbe una discontinuità di servizio, in quanto esposti al rischio di eventuali guasti e/o difetti nella funzione principale del sistema. Per le due API sono da osservare i valori elevati per la garanzia, safety, affidabilità e resilienza, in quanto si devono avere dei moduli che siano continuamente funzionanti e si comportino come previsto, per la verifica del pagamento o la generazione del sigillo.

2.3 Risk Analysis

Asset Value Assessment + Exposure Assessment			
ASSET	REQUISITO	VALORE	IMPATTO
BIGLIETTO	SR1 – Implementare dei meccanismi di controllo per assicurare CIA per il BIGLIETTO.	3	Violazione di Confidentiality per il requesteo SR1. Violazione di Integrity per il requesteo SR1. Violazione di Availability per il requesteo SR1.
EVENTO	SR2 – Implementare dei meccanismi di controllo per assicurare CIA per l'EVENTO.	3	Violazione di Confidentiality per il requesteo SR2. Violazione di Integrity per il requesteo SR2. Violazione di Availability per il requesteo SR2.
VERIFICARE/INVIARE PAGAMENTO (API)	SR3 – Implementare dei meccanismi di controllo per assicurare AAA + Safety + Reliability + Resilience per l'API PAGAMENTO.	3	Violazione di Authenticity per il requesteo SR3. Violazione di Assurance per il requesteo SR3. Violazione di Accountability per il requesteo SR3. Violazione di Safety per il requesteo SR3. Violazione di Reliability per il requesteo SR3. Violazione di Resilience per il requesteo SR3.
RILASCIARE SIGILLO (API)	SR4 – Implementare dei meccanismi di controllo per assicurare AAA + Safety + Reliability + Resilience per l'API RILASCIO SIGILLO.	3	Violazione di Authenticity per il requesteo SR4. Violazione di Assurance per il requesteo SR4. Violazione di Accountability per il requesteo SR4. Violazione di Safety per il requesteo SR4. Violazione di Reliability per il requesteo SR4. Violazione di Resilience per il requesteo SR4.
INVALIDARE BIGLIETTO	SR5 – Implementare dei meccanismi di controllo per assicurare AAA + Safety + Reliability + Resilience per INVALIDARE BIGLIETTO.	2	Violazione di Authenticity per il requesteo SR5. Violazione di Accountability per il requesteo SR5. Violazione di Safety per il requesteo SR5. Violazione di Reliability per il requesteo SR5. Violazione di Resilience per il requesteo SR5.
GESTIRE EVENTO	SR6 – Implementare dei meccanismi di controllo per assicurare AAA + Safety + Reliability + Resilience per GESTIRE EVENTO.	3	Violazione di Authenticity per il requesteo SR6. Violazione di Assurance per il requesteo SR6. Violazione di Accountability per il requesteo SR6. Violazione di Safety per il requesteo SR6. Violazione di Reliability per il requesteo SR6. Violazione di Resilience per il requesteo SR6.
CREARE BIGLIETTO	SR7 – Implementare dei meccanismi di controllo per assicurare AAA + Safety + Reliability + Resilience per CREARE BIGLIETTO.	3	Violazione di Authenticity per il requesteo SR7. Violazione di Assurance per il requesteo SR7. Violazione di Accountability per il requesteo SR7. Violazione di Safety per il requesteo SR7. Violazione di Reliability per il requesteo SR7. Violazione di Resilience per il requesteo SR7.
OTTENERE BIGLIETTO	SR8 – Implementare dei meccanismi di controllo per assicurare AAA + Safety + Reliability + Resilience per OTTENERE BIGLIETTO.	2	Violazione di Authenticity per il requesteo SR8. Violazione di Assurance per il requesteo SR8. Violazione di Accountability per il requesteo SR8. Violazione di Safety per il requesteo SR8. Violazione di Reliability per il requesteo SR8. Violazione di Resilience per il requesteo SR8.

Figura 15: Asset Value Assessment & Exposure Assessment

2.3 Risk Analysis

2.3.2 Threat Identification

In questa fase si cerca di identificare quelle che sono le minacce che insorgono una volta violati i requisiti di sicurezza. Per fare ciò si ricorre al metodo Stride in cui si associa una minaccia a ciascun obiettivo di sicurezza, per poi individuare le minacce per ciascun asset. Di seguito l'elenco delle minacce previste dal modello Stride:

1. Spoofing: violazione dell'autenticazione;
2. Tampering: violazione dell'integrità;
3. Repudiation: violazione del non ripudio, per cui un utente può negare di aver compiuto un'azione svolta;
4. Information disclosure: violazione della confidenzialità;
5. Denial of service: violazione della disponibilità;
6. Elevation of privilege: violazione dell'autorizzazione;
7. Danger: violazione della safety;
8. Unreliability: violazione dell'affidabilità;
9. Absence of resilience: violazione della resilienza.

Di seguito è riportata la tabella Fig.16 relativa all'identificazione delle minacce per ciascun asset.

Property Violated ASSET	THREAT IDENTIFICATION									
	Authentication Spoofing	Integrity Tampering	Non-repudiation Repudiation	Confidentiality Information disclosure	Availability Denial Of Service	Authorization Elevation Of Privilege	Safety Danger	Reliability Unreliability	Resilience Absence Of Resilience	
BIGLIETTO	x			x	x					
EVENTO		x		x	x					
VERIFICARE/INVIARE PAGAMENTO (API)	x		x			x	x	x	x	
RILASCIARE SIGILLO (API)	x		x			x	x	x	x	
INVALIDARE BIGLIETTO	x		x			x	x	x	x	
GESTIRE EVENTO	x		x			x	x	x	x	
CREARE BIGLIETTO	x		x			x	x	x	x	
OTTENERE BIGLIETTO	x		x			x	x	x	x	

Figura 16: Threat Identification

Tra le minacce si ha la violazione dell'integrità che potrebbe potenzialmente colpire gli asset tangibili del nostro sistema (evento e biglietto) in quanto un eventuale attacco cyber o un guasto del sistema andrebbe ad intaccare l'integrità delle informazioni. Stesso ragionamento vale per la violazione della confidenzialità e della disponibilità. Si hanno inoltre la violazione dell'autenticazione, del non ripudio, dell'autorizzazione, della safety, dell'affidabilità e della resilienza come minacce per gli assets non tangibili.

2.3 Risk Analysis

Abuse Cases diagram Questa fase permette di capire il motivo per il quale un soggetto è portato a violare un obiettivo di sicurezza. In particolare, l'Abuse Case descrive come un eventuale attaccante esterno possa compiere delle azioni per arrivare a violare uno o più obiettivi di sicurezza Fig.17.

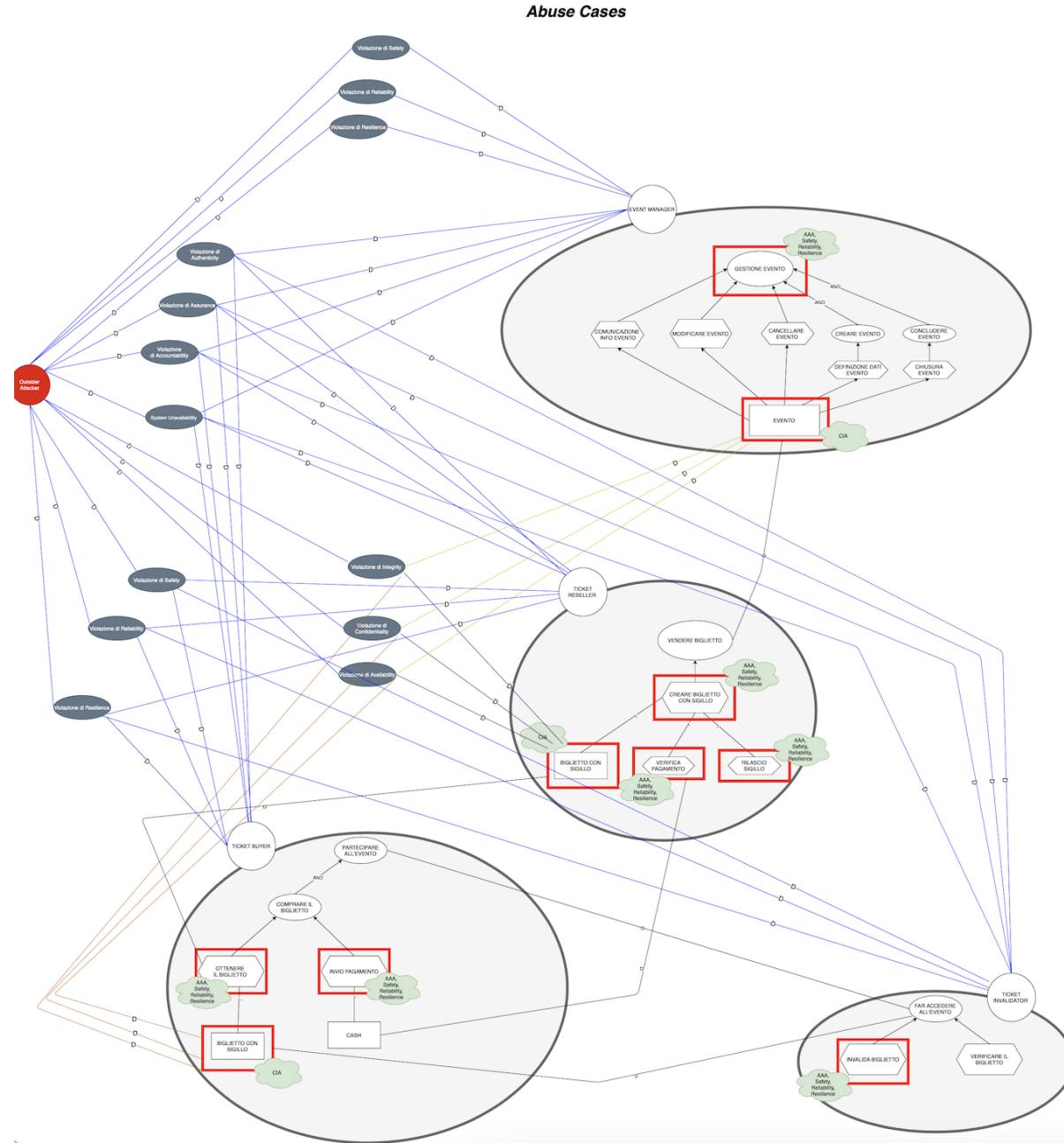


Figura 17: Abuse Case

Nota: Alcune violazioni di obiettivi di security policy (hard goal grigi) sono stati duplicati al fine di rendere maggiormente leggibili i vari collegamenti.

Misuse Cases diagram Questa sezione è importante per definire ulteriori utenti che possono causare la violazione di un obiettivo di sicurezza. Nello specifico, sono stati adottati i Misuse Case, che modellano l'interazione tra il sistema e un attore "sbadato" che potrebbe danneggiare l'infrastruttura a causa di errori compiuti involontariamente. Nella figura 18 viene riportato il diagramma *i**.

2.4 Risk Decomposition

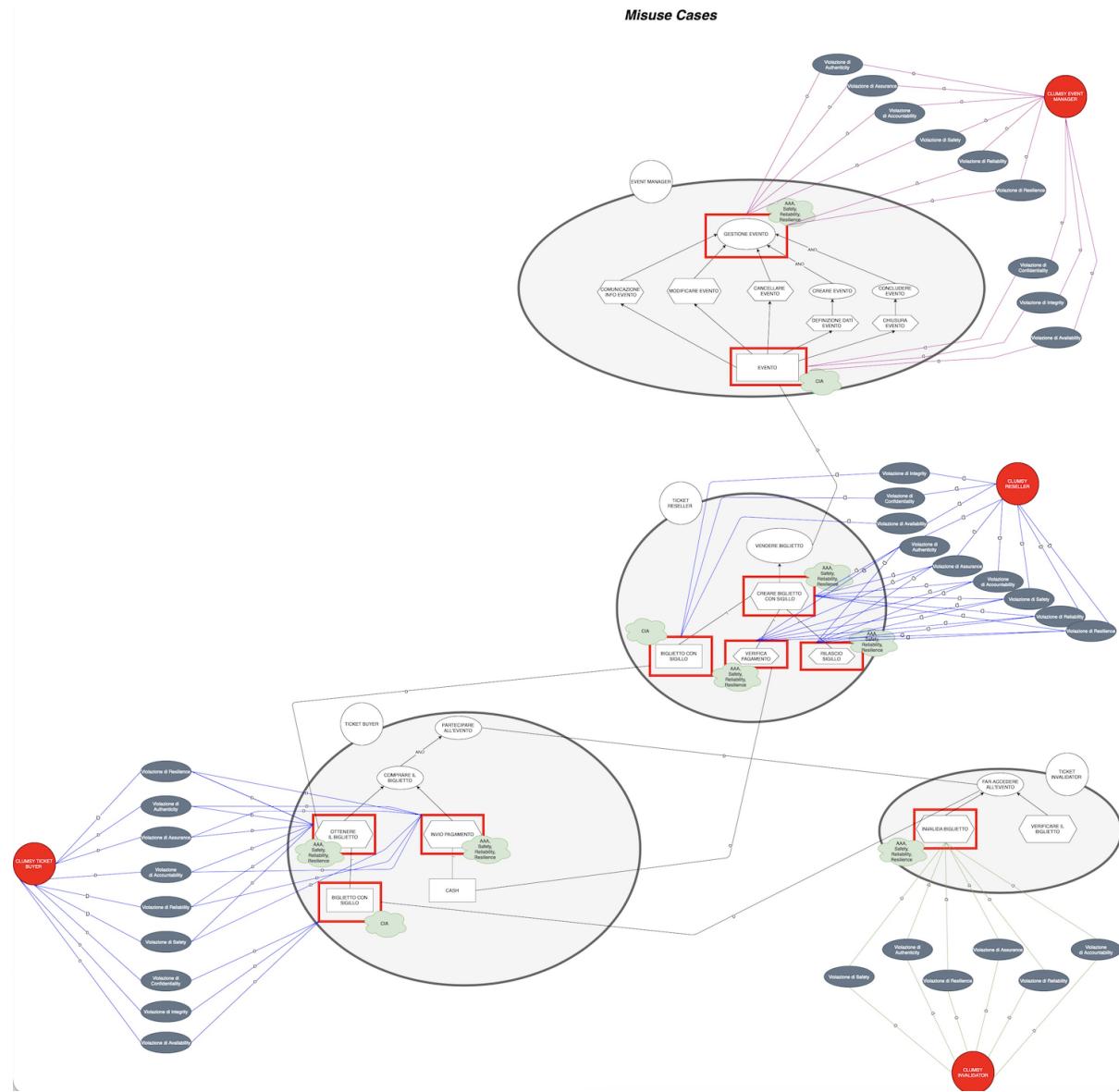


Figura 18: Misuse Case

2.4 Risk Decomposition

2.4.1 Attack Assessment

In questa fase vengono sostanzialmente analizzati gli attacchi. Nei diagrammi precedenti si sono visti tutti i possibili requisiti potenzialmente violabili da parte di attaccanti esterni o utenti disattenti. In questa sezione si ragiona invece sui possibili vettori d'attacco andando a realizzare i cosiddetti **Attack Tree**, letteralmente alberi d'attacco. In essi vengono espresse in maniera dettagliata tutte le tecniche che possono essere utilizzate e che concorrono alla violazione di un determinato requisito. Si riporta per primo (Fig.19) l'Attack Tree relativo ai casi d'abuso in cui si ipotizza che l'attaccante esterno abbia intenzioni malevoli.

Si analizza il livello inferiore dei vettori d'attacco:

- **DoS (Denial of Service)**: indica un malfunzionamento dovuto ad un attacco informatico in cui si causa lo stop delle risorse di un sistema informatico che fornisce un servizio ai client. Può causare un danneggiamento delle copie di dati o un attacco hardware che potrebbe portare ad uno shutdown dell'intero sistema violando così affidabilità e disponibilità.

2.4 Risk Decomposition

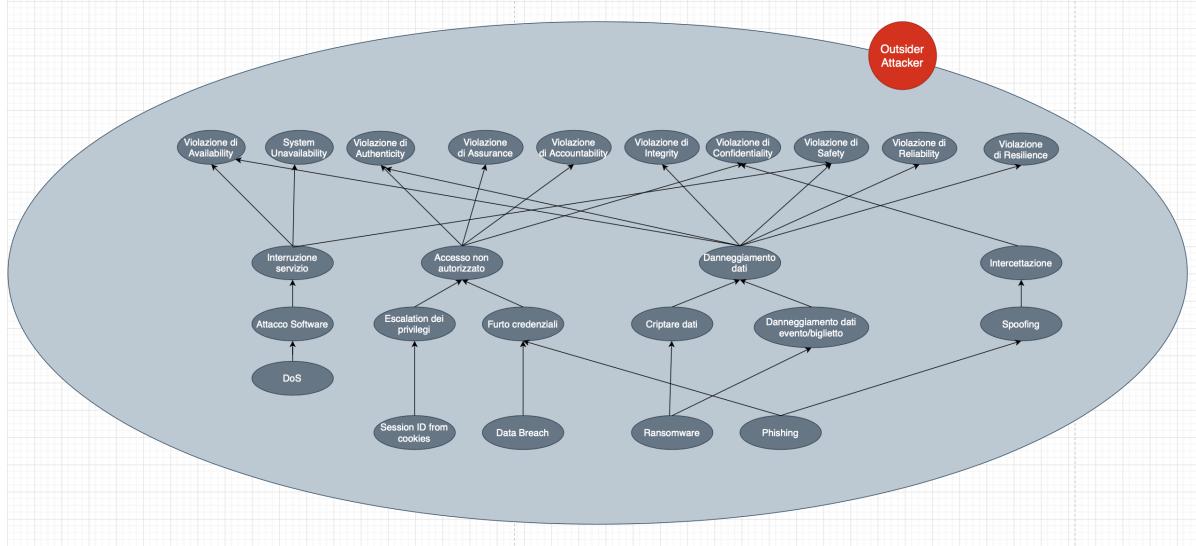


Figura 19: Attack Tree Abuse Case

- **Session ID from cookies:** consiste nel modificare i contenuti di un cookie al fine di eludere i meccanismi di sicurezza. L'attaccante può ottenere informazioni private e non autorizzate da un utente, nonché rubare l'identità digitale. Al di sopra troviamo l'escalation orizzontale dei privilegi (l'attaccante riesce a guadagnare l'accesso a risorse le quali normalmente dovrebbero essere protette da un'applicazione o da un utente). Essa fa capo all'accesso non autorizzato che viola i requisiti di confidenzialità, integrità, disponibilità, autenticità e responsabilità.
- **Data Breach:** è un incidente nella sicurezza informatica durante il quale si ha accesso a delle informazioni senza autorizzazione. Le informazioni rischiano di essere rese pubbliche senza il consenso degli utenti. Il Data Breach potrebbe essere sia causato dalle intenzioni malevoli di hacker che dall'incauto utilizzo dei dati da parte delle app terze. Si trova al di sotto del furto credenziali che fa capo all'accesso non autorizzato.
- **Ransomware:** è un tipo di malware che limita l'accesso al dispositivo infettato, chiedendo un riscatto da pagare per rimuovere la limitazione. Potrebbe essere criptati soltanto alcuni dati o anche l'intero sistema, che potrebbe venir sbloccato soltanto a seguito del pagamento. Tale minaccia è collegata al criptare dati e al danneggiare dei dati degli eventi e biglietti che fanno capo a livello superiore al danneggiamento dati. Quest'ultimo provoca la violazione di tutti e tre i requisiti della triade CIA, dell'autenticità, della garanzia, dell'affidabilità e della resilienza.
- **Phishing:** in questo caso un malintenzionato inganna la vittima, fingendosi un ente affidabile, convincendola a fornire informazioni personali, codici di accesso, dati bancari, ecc. La diretta conseguenza del Phishing sono il furto delle credenziali e lo spoofing che si attestano al di sotto dell'intercettazione e dell'accesso non autorizzato.

Si analizzano ora gli Attack Tree relativi ai misuse case in cui si ha a che fare con utenti disattentivi e non malintenzionati.

2.4 Risk Decomposition

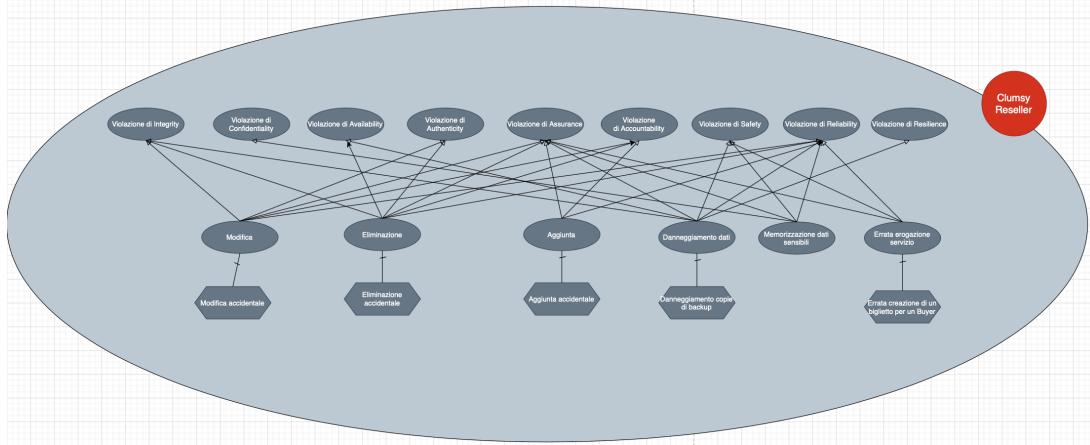


Figura 20: Clumsy Reseller

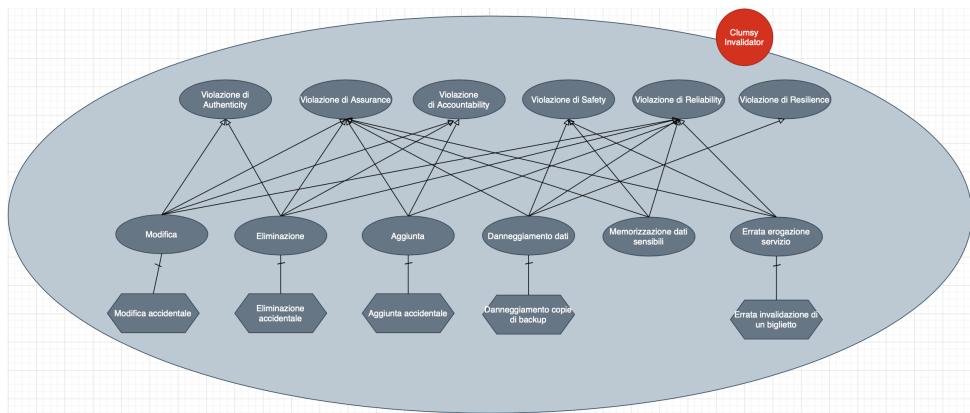


Figura 21: Clumsy Invalidator

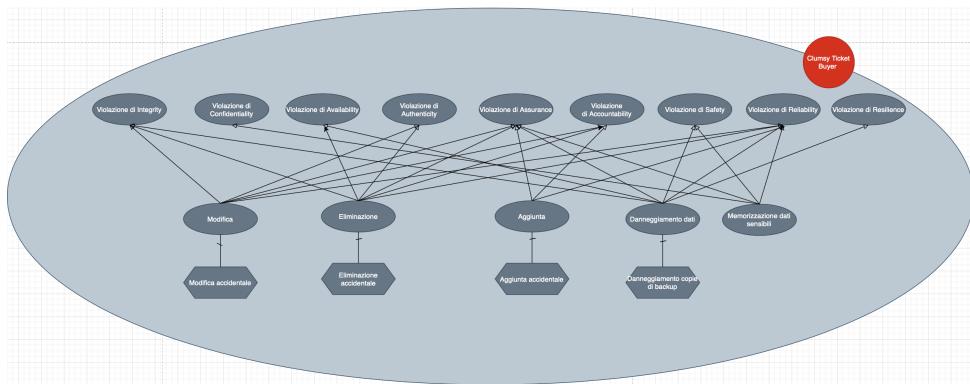


Figura 22: Clumsy Ticket Buyer

2.4 Risk Decomposition

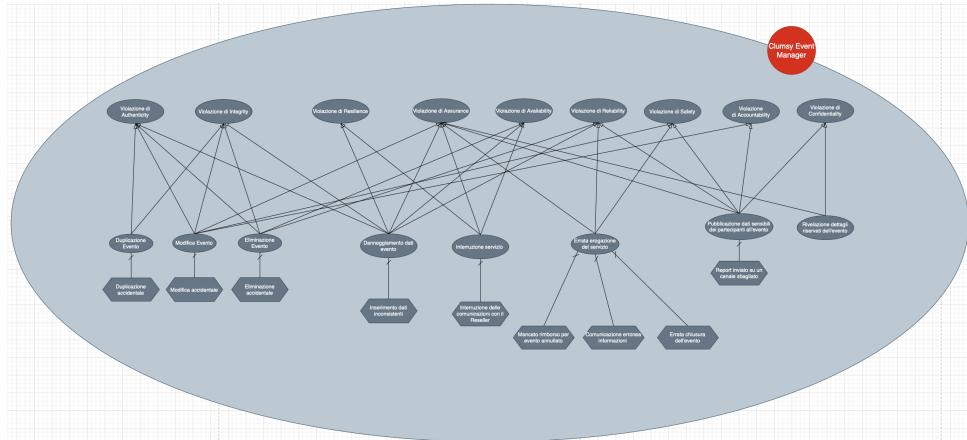


Figura 23: Clumsy Event Manager

Tabelle di Jacobson degli Attack Tree In questa sezione vengono riportate le tabelle di Jacobson principali relative agli attack tree, così da descrivere alcune delle loro componenti più nel dettaglio.

Use case Name:	Accesso non autorizzato
Actors	Outside Attacker, Event Manager
Description	Un attaccante esterno accede senza autorizzazione ai dati riservati dell'evento
Data	Evento
Stimulus and Preconditions	L'Event Manager non protegge in maniera corretta l'accesso ai dati dell'evento
Attack Flow 1	L'attaccante riesce a recuperare le credenziali dell'event manager L'attaccante modifica/elimina i dati dell'evento
Responde and Postconditions	Le informazioni dell'evento non sono più disponibili e/o veritiero
Mitigations	
Non Functional Requirements	

Figura 24: Tabella di Jacobson relativa ad Accesso non autorizzato

Use case Name:	Danneggiamento dati
Actors	Outside Attacker, Ticket Buyer, Ticket Reseller, Event Manager
Description	L'attaccante riesce a modificare i dati dell'evento e/o i dati del biglietto acquistato dal ticket buyer
Data	Biglietto, Evento
Stimulus and Preconditions	I dati dell'evento/biglietto non sono protetti correttamente.
Attack Flow 1	L'attaccante recupera le credenziali dell'event manger L'attaccante modifica i dati dell'evento
Attack Flow 2	L'attaccante si pone come "man in the middle" tra ticket reseller e buyer L'attaccante modifica i dati del biglietto
Responde and Postconditions	Il biglietto e le relative informazioni non sono più disponibili e/o veritiero.
Mitigations	
Non Functional Requirements	

Figura 25: Tabella di Jacobson relativa a Danneggiamento dati

2.4 Risk Decomposition

Preliminary Risk Assessment Report In questa fase della progettazione vengono messi in correzione gli attacchi individuati precedentemente con i requisiti di sicurezza relativi a ciascun asset del sistema. Ogni attacco è caratterizzato da una probabilità di accadimento, conoscendo poi il valore dell'impatto di non rispettare ciascun requisito di sicurezza è possibile calcolare, qualitativamente, il rischio inherente⁴ per ciascun attacco individuato.

La valutazione del rischio avviene in maniera qualitativa, risulta complesso in questa fase, infatti, fare considerazioni di tipo quantitativo. Per ovviare a questo problema ci si serve della scala di Likert a 3 valori già discussa in precedenza. Una volta, quindi, valutati probabilità e impatto in maniera qualitativa grazie all'uso di questa metodologia, sarà possibile assegnare un valore numerico anche al rischio associato, tramite una semplice moltiplicazione:

$$Rischio = Probabilità \times Impatto \quad (1)$$

Attack Assessment						
ASSET	REQUISITO	IMPATTO REQUISITO	ATTACCO	PROBABILITÀ ATTACCO	RISCHIO INERENTE	RISCHIO INERENTE TOTALE
BIGLIETTO	Violazione di Confidentiality per il requisito SR1.	3	Accesso non autorizzato	2	6	
			Danneggiamento dati	3	9	
			Intercettazione	3	9	
			Memorizzazione dati sensibili	3	9	33
	Violazione di Integrity per il requisito SR1.	3	Accesso non autorizzato	2	6	
			Danneggiamento dati	3	9	
			Modifica	2	6	
			Eliminazione	3	9	39
	Violazione di Availability per il requisito SR1.	2	Memorizzazione dati sensibili	3	9	
			Interruzione servizio	3	6	
			Danneggiamento dati	3	6	18
EVENTO	Violazione di Confidentiality per il requisito SR2.	3	Eliminazione	3	6	
			Accesso non autorizzato	2	6	
			Danneggiamento dati	3	9	
			Intercettazione	3	9	
			Rivelazione dettagli riservati dell'evento	2	6	30
	Violazione di Integrity per il requisito SR2.	3	Pubblicazione dati sensibili dei partecipanti all'evento	1	3	
			Accesso non autorizzato	2	6	
			Danneggiamento dati	3	9	
			Danneggiamento dati evento	3	9	39
VERIFICARE/INVIRE PAGAMENTO (API)	Violazione di Availability per il requisito SR2.	2	Eliminazione evento	3	9	
			Modifica evento	2	6	
			Duplicazione evento	1	3	
			Interruzione servizio	3	6	
	Violazione di Authenticity per il requisito SR3.	2	Danneggiamento dati	3	6	
			Accesso non autorizzato	2	4	
			Danneggiamento dati	3	6	
			Modifica	2	4	20
			Eliminazione	3	6	
VERIFICARE/INVIRE PAGAMENTO (API)	Violazione di Assurance per il requisito SR3.	3	Accesso non autorizzato	2	6	
			Modifica	2	6	
			Eliminazione	3	9	
			Aggiunta	2	6	
			Danneggiamento dati	3	9	51
	Violazione di Accountability per il requisito SR3.	2	Memorizzazione dati sensibili	3	9	
			Errata erogazione servizio	2	6	
			Accesso non autorizzato	2	4	
			Modifica	2	4	18
VERIFICARE/INVIRE PAGAMENTO (API)	Violazione di Safety per il requisito SR3.	3	Eliminazione	3	6	
			Aggiunta	2	4	
			Interruzione servizio	3	9	
			Danneggiamento dati	3	9	33
	Violazione di Reliability per il requisito SR3.	3	Memorizzazione dati sensibili	3	9	
			Errata erogazione servizio	2	6	
			Danneggiamento dati	3	9	
			Memorizzazione dati sensibili	3	9	
			Errata erogazione servizio	2	6	45
	Violazione di Resilience per il requisito SR3.	3	Modifica	2	6	
			Eliminazione	3	9	
			Aggiunta	2	6	
			Danneggiamento dati	3	9	9

Figura 26: Attack Assessment pt.1

⁴Con **rischio inherente** si intende il valore del rischio a monte di qualsiasi scelta di mitigazione.

2.4 Risk Decomposition

RILASCIARE SIGILLO (API)	Violazione di Authenticity per il requisito SR4.	3	Accesso non autorizzato	2	6	24			
			Danneggiamento dati	2	6				
			Modifica	2	6				
			Eliminazione	2	6				
	Violazione di Assurance per il requisito SR4.		Accesso non autorizzato	2	6	48			
			Modifica	2	6				
			Eliminazione	3	9				
			Aggiunta	2	6				
			Danneggiamento dati	3	9				
Violazione di Accountability per il requisito SR4.	Violazione di Safety per il requisito SR4.	3	Memorizzazione dati sensibili	1	3	30			
			Errata erogazione servizio	3	9				
			Accesso non autorizzato	2	4				
			Modifica	2	4				
	Violazione di Reliability per il requisito SR4.		Eliminazione	3	6	18			
			Aggiunta	2	4				
			Interruzione servizio	3	9				
			Danneggiamento dati	3	9				
			Memorizzazione dati sensibili	1	3				
Violazione di Resilience per il requisito SR4.			Errata erogazione servizio	3	9				
			Danneggiamento dati	3	9				

Figura 27: Attack Assessment pt.2

INVALIDARE BIGLIETTO	Violazione di Authenticity per il requisito SR5.	3	Accesso non autorizzato	3	9	27			
			Danneggiamento dati	2	6				
			Modifica	2	6				
			Eliminazione	2	6				
	Violazione di Assurance per il requisito SR5.		Accesso non autorizzato	3	9	54			
			Modifica	2	6				
			Eliminazione	3	9				
			Aggiunta	2	6				
			Danneggiamento dati	3	9				
GESTIRE EVENTO	Violazione di Accountability per il requisito SR5.	2	Memorizzazione dati sensibili	2	6	20			
			Errata erogazione servizio	3	9				
			Accesso non autorizzato	3	6				
			Modifica	2	4				
	Violazione di Safety per il requisito SR5.		Eliminazione	3	6				
			Aggiunta	2	4				
			Interruzione servizio	3	9				
			Danneggiamento dati	3	9				
			Memorizzazione dati sensibili	2	6				
Violazione di Reliability per il requisito SR5.			Errata erogazione servizio	3	9				
			Danneggiamento dati	3	9				
			Memorizzazione dati sensibili	2	6				
			Errata erogazione servizio	3	9				
			Modifica	2	6				
			Eliminazione	3	9				
			Aggiunta	2	6				
Violazione di Resilience per il requisito SR5.			Danneggiamento dati	3	9				
			Danneggiamento dati	3	9				
GESTIRE EVENTO	Violazione di Authenticity per il requisito SR6.	3	Accesso non autorizzato	2	6	36			
			Danneggiamento dati	3	9				
			Duplicazione evento	2	6				
			Modifica evento	2	6				
	Violazione di Assurance per il requisito SR6.		Eliminazione evento	3	9				
			Accesso non autorizzato	2	6	48			
			Modifica evento	2	6				
			Danneggiamento dati	3	9				
			Interruzione servizio	2	6				
	Violazione di Accountability per il requisito SR6.		Errata erogazione servizio	2	6	21			
			Pubblicazione dati sensibili dei partecipanti all'evento	3	9				
			Rivelazione dettagli riservati dell'evento	2	6				
			Danneggiamento dati	3	9				
Violazione di Safety per il requisito SR6.			Modifica evento	2	6	36			
			Interruzione servizio	2	6				
			Danneggiamento dati	3	9				
			Errata erogazione servizio	2	6				
			Pubblicazione dati sensibili dei partecipanti all'evento	3	9				
Violazione di Reliability per il requisito SR6.			Modifica evento	2	6	33			
			Danneggiamento dati	3	9				
			Errata erogazione servizio	2	6				
			Pubblicazione dati sensibili dei partecipanti all'evento	3	9				
Violazione di Resilience per il requisito SR6.			Eliminazione evento	3	9	15			
			Danneggiamento dati	3	9				
			Interruzione servizio	2	6				

Figura 28: Attack Assessment pt.3

2.4 Risk Decomposition

CREARE BIGLIETTO	Violazione di Authenticity per il requisito SR7.	3	Accesso non autorizzato	3	9	30
			Danneggiamento dati	2	6	
			Eliminazione	3	9	
			Modifica	2	6	
	Violazione di Assurance per il requisito SR7.	3	Accesso non autorizzato	3	9	48
			Modifica	2	6	
			Eliminazione	3	9	
			Aggiunta	1	3	
			Danneggiamento dati	2	6	
OTTENERE BIGLIETTO	Violazione di Accountability per il requisito SR7.	3	Memorizzazione dati sensibili	3	9	27
			Errata erogazione servizio	2	6	
			Accesso non autorizzato	3	9	
			Modifica	2	6	
	Violazione di Safety per il requisito SR7.	2	Eliminazione	3	9	20
			Aggiunta	1	3	
			Internuzione servizio	3	6	
			Danneggiamento dati	3	6	
			Memorizzazione dati sensibili	2	4	
OTTENERE BIGLIETTO	Violazione di Reliability per il requisito SR7.	2	Errata erogazione servizio	2	4	26
			Danneggiamento dati	3	6	
			Modifica	2	4	
			Eliminazione	3	6	
			Aggiunta	1	2	
	Violazione di Resilience per il requisito SR7.	3	Memorizzazione dati sensibili	2	4	26
			Errata erogazione servizio	2	4	
			Danneggiamento dati	3	9	
OTTENERE BIGLIETTO	Violazione di Authenticity per il requisito SR8.	2	Accesso non autorizzato	2	4	18
			Danneggiamento dati	2	4	
			Modifica	2	4	
			Eliminazione	3	6	
	Violazione di Assurance per il requisito SR8.	3	Accesso non autorizzato	2	6	33
			Modifica	2	6	
			Eliminazione	3	9	
			Aggiunta	1	3	
	Violazione di Accountability per il requisito SR8.	2	Danneggiamento dati	2	6	16
			Memorizzazione dati sensibili	1	3	
			Accesso non autorizzato	2	4	
OTTENERE BIGLIETTO	Violazione di Safety per il requisito SR8.	1	Modifica	2	4	16
			Eliminazione	3	6	
			Aggiunta	1	2	
	Violazione di Reliability per il requisito SR8.	1	Internuzione servizio	2	2	5
			Danneggiamento dati	2	2	
OTTENERE BIGLIETTO	Violazione di Resilience per il requisito SR8.	3	Memorizzazione dati sensibili	1	1	9
			Danneggiamento dati	2	2	
			Modifica	2	2	

Figura 29: Attack Assessment pt.4

2.4 Risk Decomposition

Mitigation Table parte 1 La tabella seguente, Fig. 30, rappresenta la prima parte della Mitigation Table ed ha l'obiettivo di associare i possibili attacchi che sono emersi dalle analisi precedenti ad una o più minacce individuate, per ciascun asset, grazie alla metodologia STRIDE.

Property Violated		Threat										Attack	Probability
		Authentication	Integrity	Non-repudiation	Confidentiality	Availability	Authorization	Safety	Reliability	Resilience			
ASSET		Spoofing	Tempering	Reputation	Information disclosure	Denial Of Service	Elevation Of Privilege	Danger	Irreversibility	Absence Of Resilience			
BIGLIETTO			X		X							Accesso non autorizzato	2
				X								Danneggiamento dati	3
					X							Intercettazione	3
						X						Memorizzazione dati sensibili	3
EVENTO				X								Modifica	2
					X							Eliminazione	3
						X						Interruzione servizio	3
							X					Accesso non autorizzato	2
				X				X				Danneggiamento dati	3
					X				X			Intercettazione	3
						X				X		Rivelazione dettagli riservati dell'evento	2
							X				X	Pubblicazione dati sensibili dei partecipanti all'evento	1
VERIFICARE/INVIARE PAGAMENTO (API)			X									Danneggiamento dati evento	3
				X								Eliminazione evento	3
					X							Modifica evento	2
						X						Duplicazione evento	1
				X								Interruzione servizio	3
					X							Accesso non autorizzato	2
						X						Danneggiamento dati	3
							X					Modifica	2
GENERAZIONE SIGILLO (API)			X				X					Eliminazione	2
				X				X				Aggiunta	2
					X				X			Memorizzazione dati sensibili	3
						X				X		Errata erogazione servizio	2
				X								Interruzione servizio	3
					X							Accesso non autorizzato	3
						X						Danneggiamento dati	2
							X					Modifica	2
INVALIDARE BIGLIETTO			X				X					Eliminazione	2
				X				X				Aggiunta	2
					X				X			Memorizzazione dati sensibili	2
						X				X		Errata erogazione servizio	3
				X								Accesso non autorizzato	3
					X							Danneggiamento dati	3
						X						Modifica	2
							X					Eliminazione	3
GESTIRE EVENTO			X				X					Interruzione servizio	2
				X				X				Errata erogazione servizio	2
					X				X			Rivelazione dettagli riservati dell'evento	2
						X				X		Pubblicazione dati sensibili partecipanti all'evento	3
				X								Aggiunta	1
					X							Memorizzazione dati sensibili	3
						X						Errata erogazione servizio	2
							X					Accesso non autorizzato	3
CREARE BIGLIETTO			X				X					Danneggiamento dati	2
				X				X				Modifica	2
					X				X			Eliminazione	3
						X			X			Aggiunta	1
				X					X			Memorizzazione dati sensibili	3
					X					X		Errata erogazione servizio	2
						X						Accesso non autorizzato	2
							X					Danneggiamento dati	3
OTTENERE BIGLIETTO			X				X					Modifica	2
				X				X				Eliminazione	3
					X				X			Aggiunta	1
						X			X			Memorizzazione dati sensibili	1
				X					X			Interruzione servizio	2
					X					X		Accesso non autorizzato	2
						X						Danneggiamento dati	3
							X					Modifica	2

Figura 30: Mitigation Table

2.5 Risk Reduction

Dopo aver analizzato i vari asset, valutandone in maniera generale la vulnerabilità e i potenziali vettori d'attacco, grazie alla stesura degli *attack trees* per i *misuse cases* e gli *abuse cases*, si è passati ad analizzare per ciascun asset le possibili minacce ai requisiti fondamentali di sicurezza, sfruttando la metodologia STRIDE. Ciò ha permesso di individuare per ogni minaccia i possibili attacchi, valutandone successivamente la probabilità e l'impatto di ciascuno di loro. Per ogni attacco, infatti, è stato specificato il valore della probabilità e dell'impatto; ciò ha permesso di calcolare il valore del rischio inherente. L'obiettivo di questa sezione è definire delle tecniche di controllo che riescano ad eliminare o mitigare i rischi individuati.

Innanzitutto, si individueranno una serie di misure di controllo per ciascuno dei possibili attacchi, successivamente si valuteranno, in maniera qualitativa, ognuna di queste tecniche di controllo identificate, in termini di costo, fattibilità, probabilità residua di attacco e rischio residuo⁵. Il rischio residuo è così calcolato:

$$Rischio_{Residuo} = Probabilità_{Residua} \times Impatto \quad (2)$$

Lo scopo principale di queste valutazioni è quello di poter scegliere la misura di controllo adeguata prendendo in considerazione il *risk ratio* ovvero il rapporto tra il rischio residuo specifico e il rischio massimo possibile e il *value-to-cost ratio* ovvero il rapporto tra il valore del requisito del sistema minacciato dall'attacco (riportato nell' Asset Value and Exposure Assessment) e il costo per implementare la misura di controllo. Sarà quindi possibile scrivere delle soglie numeriche per il *value-to-cost ratio* e il *risk ratio* in modo da considerare per ogni tecnica di controllo rispettivamente tre valori: "alto", "medio" o "basso". Queste considerazioni ci permetteranno la scelta di una o più misure di controllo da implementare per ogni attacco per ciascun asset considerato: si favoriranno le tecniche con valori alti del *value-to-cost ratio* e valori bassi del *risk ratio*.

2.5.1 Control Identification

In questa fase si vuole, per ciascun attacco, individuare una o più misure di controllo che possano eliminare o mitigare il rischio in termini di probabilità o impatto sfruttando, ancora una volta, la metodologia STRIDE.

Queste misure di controllo verranno poi riportate nella colonna *Control* della *Mitigation Table*. Tra le principali misure di controllo adottate nel nostro progetto, possiamo citare in particolare le ACLs, Filtering e Log.

2.5.2 Feasibility Assessment

Dopo aver individuato le tecniche di mitigazione applicabili nei diversi scenari di attacco, il passo successivo consiste nel determinare il costo e la fattibilità di ciascuna misura di controllo individuata.

Successivamente si passa alla valutazione della probabilità residua dei vari attacchi e, di conseguenza, del rischio residuo a cui si va incontro ipotizzando di adottare una determinata misura di controllo.

Le tabelle delle Figg. 31 e 32 sono strutturate in modo tale che per ogni attacco vengono riportate le misure di controllo plausibili e individuate nella fase precedente, al fine di ridurre il rischio. Accanto ad ogni misura di controllo, inoltre, è indicato il costo e un breve commento circa la fattibilità della misura stessa.

⁵Con **rischio residuo** si intende il rischio rimanente dopo aver adottato una certa misura di controllo.

2.5 Risk Reduction

Figura 31: Mitigation Table

2.5 Risk Reduction

Figura 32: Mitigation Table

2.5 Risk Reduction

Nella tabella seguente troviamo il rischio residuo per ciascuna misura di controllo. La tabella è strutturata nel modo seguente: per ogni violazione di un obiettivo di sicurezza sono riportati gli attacchi che vanno a concretizzare quella determinata minaccia, dopodiché, in corrispondenza di ogni attacco, vengono elencate le diverse tecniche di controllo individuate con il relativo costo e, infine, la probabilità residua e il rischio residuo dell'attacco sotto l'ipotesi di adottare tale misura di controllo. Nella parte destra troviamo riepilogato il rischio residuo totale per ciascuna misura di controllo e per ciascuna violazione del requisito del sistema.

Il rischio residuo totale è stato calcolato prendendo il valore massimo fra i rischi residui corrispondenti.

ASSET	REQUISITO	IMPATTO REQUISITO	ATTACCO	MISURA DI CONTROLLO	COSTO	PROBABILITÀ RESIDUA ATTACCO	RISCHIO RESIDUO	RISCHIO RESIDUO TOTALE	
								MISURA DI CONTROLLO	RISCHIO RESIDUO TOTALE
BIGLIETTO	Violazione di Confidentiality per il requisito SR1.	3	Accesso non autorizzato	ACLs	1	1	3	ACLs Encryption ACLs	9 9 6
			Intervettazione	Encryption	2	2	6		
			Memorizzazione dati sensibili	ACLs	1	1	3		
	Violazione di Integrity per il requisito SR1.	3	Accesso non autorizzato	ACLs	1	1	3	ACLs Encryption ACLs	9 3 6
			Modifica	ACLs	1	1	3		
			Eliminazione	ACLs	1	1	3		
	Violazione di Availability per il requisito SR1.	2	Memorizzazione dati sensibili	Filtering	2	2	6	ACLs Filtering Mirroring	6 10 2
			Accesso non autorizzato	Encryption	2	1	3		
			Intervettazione	Data Separation	3	2	6		
EVENTO	Violazione di Confidentiality per il requisito SR2.	3	Danneggiamento dati	ACLs	1	2	4	ACLs Encryption Data Separation	15 3 6
			Accesso non autorizzato	ACLs	1	1	3		
			Intervettazione	Encryption	2	1	3		
			Rivelazione dettagli riservati dell'evento	ACLs	1	1	3		
			Memorizzazione dati sensibili dei partecipanti all'evento	Data Separation	3	1	3		
			Pubblicazione dati sensibili dei partecipanti all'evento	Filtering	2	1	3		
	Violazione di Integrity per il requisito SR2.	3	Danneggiamento dati	ACLs	1	2	6	ACLs Filtering Tamper-resistant protocols Message authentication codes	24 9 3 3
			Accesso non autorizzato	ACLs	1	2	6		
			Danneggiamento dati	Filtering	2	1	3		
			Danneggiamento dati evento	ACLs	1	2	6		
			Eliminazione evento	Filtering	2	1	3		
			Modifica evento	ACLs	1	1	3		
VISUALIZZAZIONE	Violazione di Availability per il requisito SR2.	2	Duplicazione evento	Tamper-resistant protocols	3	1	3	ACLs Filtering Mirroring	12 10 2
			Accesso non autorizzato	Encryption	1	1	3		
			Intervettazione servizio	Filtering	2	2	4		
			Danneggiamento dati	Mirroring	3	1	2		
			Danneggiamento dati evento	ACLs	1	2	4		

Figura 33: Feasibility Assessment pt.1

2.5 Risk Reduction

ASSET	REQUISITO	IMPATTO REQUISITO	ATTACCO	MISURA DI CONTROLLO	COSTO	PROBABILITÀ RESIDUA ATTACCO	RISCHIO RESIDUO	RISCHIO RESIDUO TOTALE	
								MISURA DI CONTROLLO	RISCHIO RESIDUO TOTALE
VERIFICARE/INVVIARE PAGAMENTO (API)	SR3	Violazione di Authenticity per il requisito SR3.	Accesso non autorizzato	Log	2	1	2	Log	2
				Strong Authentication	1	1	2	Strong Authentication	2
				ACLs	1	1	2	ACLs	12
				Throttle Quotas	3	1	2	Throttle Quotas	2
				Authentication protocols	1	1	2	Authentication Protocols	4
			Danneggiamento dati	Tamper-resistant protocols	3	1	0	Tamper-Resistant Protocols	4
				Filtering	2	1	2	Filtering	2
				ACLs	1	2	4	Secure logging and auditing	2
			Modifica	Secure logging and auditing	3	1	2		
				Authentication protocols	3	1	2		
				ACLs	1	1	2		
			Eliminazione	ACLs	1	2	4		
				Filtering	2	1	2		
VERIFICARE/INVVIARE PAGAMENTO (API)	SR3	Violazione di Assurance per il requisito SR3.	Accesso non autorizzato	Log	2	1	3	Log	6
				Strong Authentication	1	1	3	Strong Authentication	3
				ACLs	1	1	3	ACLs	24
				Secure logging and auditing	3	1	3	Secure logging and auditing	3
				Authentication protocols	3	1	3	Authentication protocols	12
			Modifica	ACLs	1	1	3	Trusted third parties	3
				Authentication protocols	1	2	6	Throttle Quotas	3
				ACLs	1	2	6	Tamper-resistant protocols	3
			Eliminazione	Filtering	2	1	3	Data Separation	3
				ACLs	1	1	3		
			Aggiunta	Authentication protocols	3	1	3		
				Log	2	1	3		
				Trusted third parties	1	1	3		
				Throttle Quotas	3	1	3		
				Tamper-resistant protocols	3	1	3		
VERIFICARE/INVVIARE PAGAMENTO (API)	SR3	Violazione di Accountability per il requisito SR3.	Accesso non autorizzato	Log	2	1	2	Log	4
				Strong Authentication	1	1	2	Strong Authentication	2
				ACLs	1	1	2	ACLs	10
				Secure logging and auditing	3	1	2	Authentication protocols	4
				Authentication protocols	3	1	2	Filtering	2
			Modifica	ACLs	1	1	2	Trusted third parties	2
				ACLs	1	2	4		
				Filtering	2	1	2		
			Eliminazione	ACLs	1	2	4		
				Log	2	1	2		
			Aggiunta	Trusted third parties	1	1	2		
				Interruzione servizio	Log	2	1		
				Throttle Quotas	3	1	3		
				Authentication protocols	3	1	3		
				Tamper-resistant protocols	3	1	3		
VERIFICARE/INVVIARE PAGAMENTO (API)	SR3	Violazione di Safety per il requisito SR3.	Danneggiamento dati	Filtering	2	1	3	Throttle Quotas	3
				ACLs	1	2	6	Authentication protocols	6
				Secure logging and auditing	3	1	3	Tamper-resistant protocols	3
				Authentication protocols	3	1	3	Filtering	3
				ACLs	1	2	6	ACLs	9
			Memorizzazione dati sensibili	Secure logging and auditing	3	1	3	Data Separation	3
				Authentication protocols	3	1	3		
				ACLs	1	1	3		
			Errata erogazione servizio	Log	2	1	3		
				Trusted third parties	1	1	3		
VERIFICARE/INVVIARE PAGAMENTO (API)	SR3	Violazione di Reliability per il requisito SR3.	Memorizzazione dati sensibili	Data Separation	3	1	3	Throttle Quotas	3
				ACLs	1	1	3	Authentication protocols	12
				Authentication protocols	3	1	3	Tamper-resistant protocols	3
				ACLs	1	2	6	Filtering	6
				Secure logging and auditing	3	1	3	ACLs	21
			Modifica	Authentication protocols	3	1	3	Secure logging and auditing	3
				ACLs	1	1	3	Trusted third parties	3
				ACLs	1	2	6	Log	3
			Eliminazione	Filtering	2	1	3	Data Separation	3
				ACLs	1	2	6		
VERIFICARE/INVVIARE PAGAMENTO (API)	SR3	Violazione di Resilience per il requisito SR3.	Danneggiamento dati	Log	2	1	3	Throttle Quotas	3
				Throttle Quotas	3	1	3	Authentication protocols	3
				Authentication protocols	3	1	3	Tamper-resistant protocols	3
				Tamper-resistant protocols	3	1	3	Filtering	3
				Filtering	2	1	3	ACLs	6
			Memorizzazione dati sensibili	ACLs	1	2	6		
				Authentication protocols	3	1	3		
			Errata erogazione servizio	Log	2	1	3		
				Trusted third parties	1	1	3		

Figura 34: Feasibility Assessment pt.2

2.5 Risk Reduction

ASSET	REQUISITO	IMPATTO REQUISITO	ATTACCO	MISURA DI CONTROLLO	COSTO	PROBABILITÀ RESIDUA ATTACCO	RISCHIO RESIDUO	RISCHIO RESIDUO TOTALE	
								MISURA DI CONTROLLO	RISCHIO RESIDUO TOTALE
RILASCIARE SIGILLO (API)	Violazione di Authenticity per il requisito SR4.	3	Accesso non autorizzato, Danneggiamento dati, Modifica, Eliminazione	Log	2	1	3	Log	3
				Strong Authentication	1	1	3	Strong Authentication	3
				ACLs	1	1	3	ACLs	12
				Throttle Quotas	3	1	3	Throttle Quotas	3
				Authentication protocols	3	2	6	Authentication protocols	12
				Tamper-resistant protocol	3	3	9	Tamper-resistant protocols	9
				Filtering	2	1	3	Filtering	6
				ACLs	1	1	3	Secure logging and auditing	6
				Aggiunta	2	1	3		
RILASCIARE SIGILLO (API)	Violazione di Assurance per il requisito SR4.	3	Accesso non autorizzato, Modifica, Eliminazione, Aggiunta	Log	2	1	3	Log	9
				Strong Authentication	1	1	3	Strong Authentication	3
				ACLs	1	1	3	ACLs	18
				Secure logging and auditi	3	2	6	Secure logging and auditing	6
				Authentication protocols	3	2	6	Authentication protocols	21
				ACLs	1	1	3	Filtering	6
				ACLs	1	1	3	Trusted third parties	3
				Eliminazione	2	1	3	Throttle Quotas	3
				Filtering	2	1	3	Tamper-resistant protocols	9
				ACLs	1	1	3	Data Separation	6
RILASCIARE SIGILLO (API)	Violazione di Accountability per il requisito SR4.	2	Accesso non autorizzato, Modifica, Eliminazione, Aggiunta	Log	2	1	2	Log	6
				Strong Authentication	1	1	2	Strong Authentication	2
				ACLs	1	1	2	ACLs	8
				Secure logging and auditi	3	2	4	Secure logging and auditing	4
				Authentication protocols	3	2	4	Authentication protocols	6
				ACLs	1	1	2	Filtering	2
				Eliminazione	2	1	2	Trusted third parties	2
				Filtering	2	1	2		
				ACLs	1	1	2		
				Aggiunta	3	1	2		
RILASCIARE SIGILLO (API)	Violazione di Safety per il requisito SR4.	3	Accesso non autorizzato, Modifica, Eliminazione, Aggiunta	Log	2	1	2	Log	3
				Strong Authentication	1	1	2	Throttle Quotas	3
				ACLs	1	1	2	Authentication protocols	12
				Secure logging and auditi	3	2	6	Tamper-resistant protocols	9
				Authentication protocols	3	2	6	Filtering	3
				ACLs	1	1	2	ACLs	6
				Eliminazione	2	1	3	Data Separation	6
				Filtering	2	1	3		
				ACLs	1	1	3		
				Aggiunta	3	2	6		
RILASCIARE SIGILLO (API)	Violazione di Reliability per il requisito SR4.	3	Accesso non autorizzato, Modifica, Eliminazione, Aggiunta	Log	2	1	2	Throttle Quotas	3
				Strong Authentication	1	1	2	Authentication protocols	21
				ACLs	1	1	2	Tamper-resistant protocols	9
				Secure logging and auditi	3	2	6	Filtering	6
				Authentication protocols	3	2	6	ACLs	15
				ACLs	1	1	3	Secure logging and auditing	6
				Eliminazione	2	1	3	Log	6
				Filtering	2	1	3	Trusted third parties	3
				ACLs	1	1	3	Data Separation	6
				Aggiunta	3	2	6		
RILASCIARE SIGILLO (API)	Violazione di Resilience per il requisito SR4.	3	Accesso non autorizzato, Modifica, Eliminazione, Aggiunta	Log	2	1	2	Throttle Quotas	3
				Strong Authentication	1	1	2	Authentication protocols	6
				ACLs	1	1	2	Tamper-resistant protocols	9
				Secure logging and auditi	3	2	6	Filtering	3
				Authentication protocols	3	2	6	ACLs	3
				Eliminazione	2	1	3		
				Filtering	2	1	3		
				ACLs	1	1	3		
				Aggiunta	3	2	6		

Figura 35: Feasibility Assessment pt.3

2.5 Risk Reduction

ASSET	REQUISITO	IMPATTO REQUISITO	ATTACCO	MISURA DI CONTROLLO	COSTO	PROBABILITÀ RESIDUA ATTACCO	RISCHIO RESIDUO	RISCHIO RESIDUO TOTALE		
								MISURA DI CONTROLLO	RISCHIO RESIDUO TOTALE	
INVALIDARE BIGLIETTO	Violazione di Authenticity per il requisito SRS.	3		Accesso non autorizzato	Log	2	1	3		
				Strong Authentication	1	1	3	Log	3	
				ACLs	1	1	3	Strong Authentication	3	
				Throttle Quotas	3	1	3	ACLs	12	
				Authentication protocols	3	2	6	Throttle Quotas	3	
				Tamper-resistant protocol	3	3	9	Authentication protocols	12	
				Filtering	2	1	3	Tamper-resistant protocols	9	
				ACLs	1	1	3	Filtering	6	
				Eliminazione	Filtering	2	1	3	Secure logging and auditing	6
INVALIDARE BIGLIETTO	Violazione di Assurance per il requisito SRS.	3		Accesso non autorizzato	Log	2	1	3		
				Strong Authentication	1	1	3	Log	9	
				ACLs	1	1	3	Strong Authentication	3	
				Secure logging and auditi	3	2	6	ACLs	18	
				Modifica	Authentication protocols	3	2	6	Secure logging and auditing	6
				ACLs	1	1	3	Authentication protocols	21	
				Eliminazione	ACLs	1	1	3	Filtering	6
				Filtering	2	1	3	Trusted third parties	3	
				ACLs	1	1	3	Throttle Quotas	3	
				Aggiunta	Authentication protocols	3	1	3	Tamper-resistant protocols	9
INVALIDARE BIGLIETTO	Violazione di Accountability per il requisito SRS.	2		Accesso non autorizzato	Log	2	1	2	Data Separation	6
				Strong Authentication	1	1	2			
				ACLs	1	1	2	Log	6	
				Secure logging and auditi	3	2	4	Strong Authentication	2	
				Modifica	Authentication protocols	3	2	4	ACLs	8
				ACLs	1	1	2	Secure logging and auditing	4	
				Eliminazione	Filtering	2	1	2	Authentication protocols	6
				ACLs	1	1	2	Filtering	2	
				Aggiunta	Authentication protocols	3	1	2	Trusted third parties	2
				Log	2	2	4			
INVALIDARE BIGLIETTO	Violazione di Safety per il requisito SRS.	3		Interruzione servizio	Log	2	1	3		
				Throttle Quotas	3	1	3	Log	3	
				Authentication protocols	3	2	6	Throttle Quotas	3	
				Danneggiamento dati	Temper-resistant protocol	3	3	9	Authentication protocols	12
				Filtering	2	1	3	Temper-resistant protocols	9	
				ACLs	1	1	3	Filtering	3	
				Memorizzazione dati sensibili	Data Separation	3	2	6	ACLs	6
				Errata erogazione servizio	ACLs	1	1	3	Data Separation	6
				Aggiunta	Authentication protocols	3	2	6		
				Throttle Quotas	3	1	3			
INVALIDARE BIGLIETTO	Violazione di Reliability per il requisito SRS.	3		Danneggiamento dati	Authentication protocols	3	2	6	Throttle Quotas	3
				Temper-resistant protocol	3	3	9	Authentication protocols	21	
				Filtering	2	1	3	Temper-resistant protocols	9	
				ACLs	1	1	3	Filtering	6	
				Modifica	Secure logging and auditi	3	2	6	ACLs	15
				ACLs	1	1	3	Secure logging and auditing	6	
				Eliminazione	Authentication protocols	3	2	6	Log	6
				ACLs	1	1	3	Trusted third parties	3	
				Filtering	2	1	3	Data Separation	6	
				Aggiunta	Authentication protocols	3	1	3		
INVALIDARE BIGLIETTO	Violazione di Resilience per il requisito SRS.	3		Danneggiamento dati	Throttle Quotas	3	1	3	Throttle Quotas	3
				Authentication protocols	3	2	6	Authentication protocols	6	
				Temper-resistant protocol	3	3	9	Temper-resistant protocols	9	
				Filtering	2	1	3	Filtering	3	
				ACLs	1	1	3	ACLs	3	
				Memorizzazione dati sensibili	Data Separation	3	2	6		
				Errata erogazione servizio	ACLs	1	1	3		
				Aggiunta	Authentication protocols	3	2	6		
				Throttle Quotas	3	1	3			

Figura 36: Feasibility Assessment pt.4

2.5 Risk Reduction

ASSET	REQUISITO	IMPATTO REQUISITO	ATTACCO	MISURA DI CONTROLLO	COSTO	PROBABILITÀ RESIDUA ATTACCO	RISCHIO RESIDUO	RISCHIO RESIDUO TOTALE	
								MISURA DI CONTROLLO	RISCHIO RESIDUO TOTALE
GESTIRE EVENTO	Violazione di Authenticity per il requisito SR6.	3	Danneggiamento dati	Accesso non autorizzato	Log Strong Authentication ACLs	2 1 1	1 3 3	3	3
			Danneggiamento dati	Throttle Quotas Authentication protocols Tamper-resistant protocols	3 3 3	1 1 1	3 3 3	3	18
			Duplicazione evento	Filtering ACLs	2 1	1 2	3 6	6	6
			Modifica evento	Message authentication codes ACLs	1 1	1 1	3 3	3	3
			Eliminazione evento	ACLs	1	2	6	6	6
			Danneggiamento dati	Filtering	2	1	3	3	3
			Accesso non autorizzato	Log Strong Authentication ACLs	2 1 1	1 3 3	3	3	3
			Modifica evento	ACLs	1	2	6	6	6
			Danneggiamento dati	Throttle Quotas Authentication protocols Tamper-resistant protocols	3 3 3	1 1 1	3 3 3	3	9
			Errata ergazione servizio	Filtering ACLs	2 1	1 2	3 6	6	6
GESTIRE EVENTO	Violazione di Assurance per il requisito SR6.	3	Danneggiamento dati	Authentication protocol Tamper-resistant protocols	3 3	1 1	3 3	3	3
			Pubblicazione dati sensibili dei partecipanti all'evento	Authentication protocols ACLs	3 1	2 1	6 3	6	6
			Rivelazione dettagli riservati dell'evento	Strong Authentication Running applications Filtering	1 2	1 1	3 3	3	3
			Accesso non autorizzato	Log Strong Authentication ACLs	2 1 1	1 3 3	3	3	3
			Modifica evento	ACLs	1	2	6	6	6
			Danneggiamento dati	Throttle Quotas Authentication protocols Tamper-resistant protocols	3 3 3	1 1 1	3 3 3	3	9
			Errata ergazione servizio	Filtering ACLs	2 1	1 2	3 6	6	6
			Pubblicazione dati sensibili dei partecipanti all'evento	Authentication protocols ACLs	3 1	1 2	6 6	6	6
			Modifica evento	Strong Authentication ACLs	1 1	1 1	3 3	3	3
			Interruzione servizio	Log Throttle Quotas Authentication protocols	2 3 3	1 1 1	3 3 3	3	3
GESTIRE EVENTO	Violazione di Safety per il requisito SR6.	3	Danneggiamento dati	Throttle Quotas Authentication protocols Tamper-resistant protocols	3 3 3	1 1 1	3 3 3	3	3
			Pubblicazione dati sensibili dei partecipanti all'evento	Authentication protocols ACLs	3 1	1 2	6 6	6	6
			Modifica evento	Strong Authentication ACLs	1 1	1 2	6 6	6	6
			Danneggiamento dati	Throttle Quotas Authentication protocols Tamper-resistant protocols	3 3 3	1 1 1	3 3 3	3	3
			Errata ergazione servizio	Filtering ACLs	2 1	1 2	3 6	6	6
			Pubblicazione dati sensibili dei partecipanti all'evento	Authentication protocols Strong Authentication	3 1	1 1	3 3	3	3
			Modifica evento	ACLs	1	2	6	6	6
			Danneggiamento dati	Filtering	2	1	3	3	3
			Accesso non autorizzato	Throttle Quotas Authentication protocols ACLs	3 3 3	1 1 1	3 3 3	3	3
			Modifica evento	ACLs	1	2	6	6	6
GESTIRE EVENTO	Violazione di Reliability per il requisito SR6.	3	Danneggiamento dati	Throttle Quotas Authentication protocols Tamper-resistant protocols	3 3 3	1 1 1	3 3 3	3	3
			Pubblicazione dati sensibili dei partecipanti all'evento	Authentication protocols Strong Authentication	3 1	1 1	3 3	3	3
			Modifica evento	ACLs	1	2	6	6	6
			Danneggiamento dati	Throttle Quotas Authentication protocols Tamper-resistant protocols	3 3 3	1 1 1	3 3 3	3	3
			Errata ergazione servizio	Filtering ACLs	2 1	1 2	3 6	6	6
			Pubblicazione dati sensibili dei partecipanti all'evento	Authentication protocols Strong Authentication	3 1	1 1	3 3	3	3
			Modifica evento	ACLs	1	2	6	6	6
			Danneggiamento dati	Filtering	2	1	3	3	3
			Accesso non autorizzato	Throttle Quotas Authentication protocols ACLs	3 3 3	1 1 1	3 3 3	3	3
			Modifica evento	ACLs	1	2	6	6	6
GESTIRE EVENTO	Violazione di Resilience per il requisito SR6.	3	Danneggiamento dati	Throttle Quotas Authentication protocols Tamper-resistant protocols	3 3 3	1 1 1	3 3 3	3	3
			Pubblicazione dati sensibili dei partecipanti all'evento	Authentication protocols Strong Authentication	3 1	1 1	3 3	3	3
			Modifica evento	ACLs	1	2	6	6	6
			Danneggiamento dati	Throttle Quotas Authentication protocols ACLs	3 3 3	1 1 1	3 3 3	3	3
			Errata ergazione servizio	Filtering ACLs	2 1	1 2	3 6	6	6
			Pubblicazione dati sensibili dei partecipanti all'evento	Authentication protocols Strong Authentication	3 1	1 1	3 3	3	3
			Modifica evento	ACLs	1	2	6	6	6
			Danneggiamento dati	Filtering	2	1	3	3	3
			Accesso non autorizzato	Throttle Quotas Authentication protocols ACLs	3 3 3	1 1 1	3 3 3	3	3
			Modifica evento	ACLs	1	2	6	6	6

Figura 37: Feasibility Assessment pt.5

2.5 Risk Reduction

ASSET	REQUISITO	IMPATTO REQUISITO	ATTACCO	MISURA DI CONTROLLO	COSTO	PROBABILITÀ RESIDUA ATTACCO	RISCHIO RESIDUO	RISCHIO RESIDUO TOTALE		
								MISURA DI CONTROLLO	RISCHIO RESIDUO TOTALE	
CREARE BIGLIETTO	Violazione di Authenticity per il requisito SR7.	3	Danneggiamento dati	Accesso non autorizzato	Log Strong Authentication ACLs Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering ACLs	2 1 1 3 1 3 2 1	1 1 1 1 2 1 2 1	3 3 3 3 3 3 6 3	Log Strong Authentication ACLs Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering Secure logging and auditing	3 3 18 3 6 3 6 3
			Eliminazione	Modifica	Log ACLs Filtering Secure logging and auditing Authentication protocols ACLs	1 1 2 3 3 1	2 1 1 1 1 1	6 3 3 3 3 3	Log ACLs Filtering Secure logging and auditing	24 12 6 3
			Modifica	Aggiunta	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols ACLs Tamper-resistant protocols Filtering ACLs	1 1 1 3 1 1 3 2 1	1 1 1 1 1 1 1 1 1	3 3 3 3 3 3 3 3 3	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols Tamper-resistant protocols Data Separation	5 3 24 3 3 3 3
			Modifica	Danneggiamento dati	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols ACLs Tamper-resistant protocols Filtering ACLs	1 1 1 3 1 1 3 2 1	1 1 1 1 1 1 1 1 1	3 3 3 3 3 3 3 3 3	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols Tamper-resistant protocols Data Separation	6 3 24 3 3 3 3
			Modifica	Memorizzazione dati sensibili	Log Strong Authentication ACLs Data Separation ACLs	1 1 1 3 1	1 1 1 1 1	6 3 3 3 3	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols Tamper-resistant protocols Data Separation	15 6 12 3 3 3
			Modifica	Errata erogazione servizio	Log Strong Authentication ACLs Authentication protocols ACLs Trusted third parties	1 1 1 3 1 1	1 1 1 1 1 1	3 3 3 3 3 3	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols Tamper-resistant protocols Trusted third parties	3 3 3 3 3 3
			Eliminazione	Violazione di Accountability per il requisito SR7.	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols ACLs Tamper-resistant protocols Filtering ACLs	1 1 1 3 1 1 3 2 1	1 1 1 1 1 1 1 1 1	6 3 3 3 3 3 3 6 3	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols Tamper-resistant protocols Trusted third parties	6 3 15 3 6 3
			Aggiunta	Violazione di Safety per il requisito SR7.	Log Strong Authentication ACLs Authentication protocols ACLs Tamper-resistant protocols Filtering ACLs Data Separation ACLs Authentication protocols	1 1 1 3 1 1 3 2 1 3	1 1 1 1 1 1 1 1 1 1	3 3 3 3 3 3 3 3 3 3	Log Strong Authentication ACLs Authentication protocols Tamper-resistant protocols Filtering ACLs Data Separation	2 2 2 2 2 2 2
			Aggiunta	Violazione di Reliability per il requisito SR7.	Log Strong Authentication ACLs Authentication protocols ACLs Tamper-resistant protocols Filtering ACLs Data Separation ACLs Authentication protocols	1 1 1 3 1 1 3 2 1 3	1 1 1 1 1 1 1 1 1 1	2 2 2 2 2 2 2 2 2 2	Log Strong Authentication ACLs Authentication protocols Tamper-resistant protocols Filtering ACLs Data Separation	8 8 4 2 14 2 2
			Aggiunta	Violazione di Resilience per il requisito SR7.	Log Strong Authentication ACLs Authentication protocols ACLs Tamper-resistant protocols Filtering ACLs	1 1 1 3 1 1	1 1 1 1 1 1	2 2 2 3 2 2	Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering ACLs Data Separation	3 3 3 3 6

Figura 38: Feasibility Assessment pt.6

2.5 Risk Reduction

ASSET	REQUISITO	IMPATTO REQUISITO	ATTACCO	MISURA DI CONTROLLO	COSTO	PROBABILITÀ RESIDUA ATTACCO	RISCHIO RESIDUO	RISCHIO RESIDUO TOTALE	
								MISURA DI CONTROLLO	RISCHIO RESIDUO TOTALE
OTTENERE BIGLIETTO	Violazione di Authenticity per il requisito SR8.	2	Danneggiamento dati	Accesso non autorizzato	Log	2	1	2	
				Strong Authentication	1	1	2		
				ACLs	1	1	2		
				Throttle Quotas	3	1	2		
				Authentication protocols	3	1	2		
		3	Modifica	Tamper-resistant protocols	3	1	2		
				Filtering	2	1	2		
				Secure logging and auditing	3	1	2		
				Authentication protocols	3	1	2		
				ACLs	1	1	2		
OTTENERE BIGLIETTO	Violazione di Assurance per il requisito SR8.	2	Eliminazione	ACLs	1	2	4		
				Filtering	2	1	2		
				Log	2	1	3		
				Strong Authentication	1	1	3		
				ACLs	1	1	3		
		3	Aggiunta	Secure logging and auditing	3	1	3		
				Authentication protocols	3	1	3		
				Log	2	1	3		
				Trusted third parties	1	1	3		
				Throttle Quotas	3	1	3		
OTTENERE BIGLIETTO	Violazione di Accountability per il requisito SR8.	2	Danneggiamento dati	Authentication protocols	3	1	3		
				Tamper-resistant protocols	3	1	2		
				Filtering	2	1	2		
				ACLs	1	2	4		
			Eliminazione	ACLs	1	2	2		
		1	Aggiunta	Filtering	2	1	2		
				Authentication protocols	3	1	2		
				Log	2	1	2		
				Trusted third parties	1	1	2		
			Interruzione servizio	Log	2	1	1		
OTTENERE BIGLIETTO	Violazione di Safety per il requisito SR8.	1	Danneggiamento dati	Throttle Quotas	3	1	1		
				Authentication protocols	3	1	1		
				Tamper-resistant protocols	3	1	1		
				Filtering	2	1	1		
				ACLs	1	2	2		
		1	Memorizzazione dati sensibili	Secure logging and auditing	3	1	1		
				Authentication protocols	3	1	1		
				Log	2	1	1		
				Trusted third parties	1	1	1		
			Data Separation	3	1	1			
OTTENERE BIGLIETTO	Violazione di Reliability per il requisito SR8.	1	Danneggiamento dati	Throttle Quotas	3	1	1		
				Authentication protocols	3	1	1		
				Tamper-resistant protocols	3	1	1		
				Filtering	2	1	1		
				ACLs	1	2	2		
		1	Modifica	Secure logging and auditing	3	1	1		
				Authentication protocols	3	1	1		
				Log	2	1	1		
				Trusted third parties	1	1	1		
			Data Separation	3	1	1			
OTTENERE BIGLIETTO	Violazione di Resilience per il requisito SR8.	3	Danneggiamento dati	Throttle Quotas	3	1	3		
				Authentication protocols	3	1	3		
				Tamper-resistant protocols	3	1	3		
				Filtering	2	1	3		
				ACLs	1	2	6		
		1	Aggiunta	Throttle Quotas	3	1	3		
				Authentication protocols	3	1	3		
				Log	2	1	1		
				Trusted third parties	1	1	1		
			Data Separation	3	1	1			

Figura 39: Feasibility Assessment pt.7

2.5 Risk Reduction

Tabelle di Jacobson degli Attack Tree aggiornate Di seguito (Figg. 40 e 41) vengono riportate le tabelle di Jacobson aggiornate per gli *attack tree*. In queste tabelle sono state inserite, a seguito delle analisi precedenti sulle misure di controllo, informazioni nelle ultime due righe, cioè sono state specificate le tecniche di mitigazione e i requisiti non funzionali.

Use case ID:	
Use case Name:	Accesso non autorizzato
Actors	Outside Attacker, Event Manager
Description	Un attaccante esterno accede senza autorizzazione ai dati riservati dell'evento
Data	Evento
Stimulus and Preconditions	L' Event Manager non protegge in maniera corretta l'accesso ai dati dell'evento
Attack Flow 1	L'attaccante riesce a recuperare le credenziali dell' event manager L'attaccante modifica/elimina i dati dell'evento
Responde and Postconditions	Le informazioni dell'evento non sono più disponibili e/o veritieri Le password di accesso degli utenti vengono memorizzate dopo aver applicato un algoritmo di hashing (non in chiaro). Le azioni di modifica vengono salvate sui file di log.
Mitigations	Utilizzare un pacchetto per effettuare l'hashing di un testo in chiaro. Utilizzare un pacchetto per il logging.
Non Functional Requirements	

Figura 40: Tabella di Jacobson aggiornata relativa ad Accesso non autorizzato

Use case ID:	
Use case Name:	Danneggiamento dati
Actors	Outside Attacker, Ticket Buyer, Ticket Reseller, Event Manager
Description	L'attaccante riesce a modificare i dati dell'evento e/o i dati del biglietto acquistato dal ticket buyer
Data	Biglietto, Evento
Stimulus and Preconditions	I dati dell'evento/biglietto non sono protetti correttamente.
Attack Flow 1	L' attaccante recupera le credenziali dell'event manger L'attaccante modifica i dati dell'evento
Attack Flow 2	L'attaccante si pone come "man in the middle" tra ticket reseller e buyer L'attaccante modifica i dati del biglietto
Responde and Postconditions	Il biglietto e le relative informazioni non sono più disponibili e/o veritieri. Le password di accesso degli utenti vengono memorizzate dopo aver applicato un algoritmo di hashing (non in chiaro). Le azioni di modifica vengono salvate sui file di log.
Mitigations	Utilizzare un pacchetto per effettuare l'hashing di un testo in chiaro. Utilizzare un pacchetto per il logging.
Non Functional Requirements	

Figura 41: Tabella di Jacobson aggiornata relativa a Danneggiamento dati

2.5.3 Security Requirement Definition

L'obiettivo di questa fase è la specifica dei requisiti di sicurezza.

Value-to-Cost Ratios Il primo passo è calcolare i *value-to-cost ratios* per ogni requisito di sicurezza degli asset, così da individuare i requisiti che hanno più valore e sono meno costosi.

Il rapporto *value-to-cost* è definito come il rapporto tra il valore attribuito ad un requisito di sicurezza e il costo necessario ad implementare tale requisito attraverso una specifica misura di mitigazione; da qui è possibile intuire come tale indice debba essere calcolato per ogni coppia requisito-misura di controllo. Un rapporto valore-costo elevato è sinonimo di un ampio margine tra il valore assegnato al requisito a fronte del costo, dunque la scelta di mitigazione si candida a tutti gli effetti come plausibile. Al contrario, rapporti bassi per tale indice segnalano che il costo complessivo previsto per applicare una certa misura di controllo è elevato tenendo conto del valore del corrispondente requisito.

Anche in questo caso è stata utilizzata una scala di Likert a 3 valori.

Il costo complessivo per ciascuna misura di controllo è stato calcolato come il costo massimo per la corrispondente misura.

2.5 Risk Reduction

REQUISITO	VALORE REQUISITO	COSTO TOTALE		
		MISURA DI CONTROLLO	VALORE COSTO TOTALE *	VALUE-TO-COST RATIO
SR1	3	ACLs	1	3
		Filtering	2	1,5
		Mirroring	3	1
		Encryption	2	1,5
		Data Separation	3	1
SR2	3	ACLs	1	3
		Filtering	2	1,5
		Mirroring	3	1
		Encryption	2	1,5
		Data Separation	3	1
SR3	3	Tamper-resistant protocols	3	1
		Message authentication codes	1	3
		Log	2	1,5
		Strong Authentication	1	3
		ACLs	1	3
SR4	3	Throttle Quotas	3	1
		Authentication protocols	3	1
		Tamper-resistant protocols	3	1
		Filtering	2	1,5
		Secure logging and auditing	3	1
SR5	2	Trusted third parties	1	3
		Data Separation	3	1
		Log	2	1
		Strong Authentication	1	2
		ACLs	1	2
SR6	3	Throttle Quotas	3	0,67
		Authentication protocols	3	0,67
		Tamper-resistant protocols	3	0,67
		Filtering	2	1
		Secure logging and auditing	3	0,67
SR7	3	Trusted third parties	1	2
		Data Separation	3	0,67
		Log	2	1,5
		Strong Authentication	1	3
		ACLs	1	3
SR8	2	Throttle Quotas	3	1
		Authentication protocols	3	1
		Tamper-resistant protocols	3	0,67
		Filtering	2	1
		Secure logging and auditing	3	0,67

Cost			
Value-to-cost ratio	1	2	3
1	1	0,5	0,333333333333
2	2	1	0,666666666667
3	3	1,5	1

High		
Medium		
Low		
Value	3	2

NOTE: * Preso come MAX dei costi della misura, considerando tutti gli attacchi del requisito corrispondente

Figura 42: Value to cost Ratios

Prioritize Requirements A questo punto della progettazione le informazioni note riguardo un determinato requisito sono molteplici: il rischio inherente, i rischi residui dovuti a certe scelte di mitigazione alternative ed i corrispondenti costi.

2.5 Risk Reduction

Il passo successivo da intraprendere consiste in una prioritizzazione dei requisiti al fine di capire su quali di questi convenga soffermarsi.

A tal proposito, nella tabella di Fig.43 viene eseguito il rapporto tra il rischio residuo associato ad ogni singola misura di controllo e il massimo rischio residuo che si può ottenere per un determinato requisito di sicurezza. Dunque, sono da ritenere più efficaci le scelte legate ad un basso valore per il rapporto considerato.

REQUISITO	MISURA DI CONTROLLO	VALORE RISCHIO RESIDUO TOTALE	RISCHIO RESIDUO MAX	RISCHIO %
SR1	Violazione di Confidentiality per il requisito SR1.	ACLs Encryption Data Separation	5 9 6	22,22% 33,33% 22,22%
	Violazione di Integrity per il requisito SR1.	ACLs Encryption Data Separation	9 3 6	25,00% 8,33% 16,67%
	Violazione di Availability per il requisito SR1.	ACLs Filtering Mirroring	6 10 2	22,22% 37,04% 7,41%
SR2	Violazione di Confidentiality per il requisito SR2.	ACLs Filtering Encryption Data Separation	15 6 3 6	41,67% 16,67% 8,33% 16,67%
	Violazione di Integrity per il requisito SR2.	ACLs Filtering Tamper-resistant protocols Message authentication codes	24 9 3 3	44,44% 16,67% 5,56% 5,56%
	Violazione di Availability per il requisito SR2.	ACLs Filtering Mirroring	12 10 2	33,33% 27,78% 5,56%
SR3	Violazione di Confidentiality per il requisito SR3.	Log Strong Authentication ACLs Throttle Quotas Authentication protocols Filtering Session logging and auditing	2 2 12 2 4 4 6	5,56% 5,56% 33,33% 5,56% 11,11% 11,11% 9,52%
	Violazione di Integrity per il requisito SR3.	Log Strong Authentication ACLs Throttle Quotas Authentication protocols Trusted third parties Tamper-resistant protocols Data Separation	24 3 12 3 3 3 3	38,10% 4,76% 19,05% 4,76% 4,76% 4,76% 4,76%
	Violazione di Assurance per il requisito SR3.	Log Strong Authentication ACLs Authentication protocols Filtering Trusted third parties Session logging and auditing	6 2 10 4 6 2	9,52% 5,56% 27,78% 11,11% 9,52% 5,56%
	Violazione di Accountability per il requisito SR3.	Log Strong Authentication ACLs Authentication protocols Filtering Trusted third parties	4 2 10 4 2	11,11% 5,56% 11,11% 5,56% 5,56%
	Violazione di Safety per il requisito SR3.	Log Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering ACLs Data Separation	3 3 6 3 3 9 3	8,33% 8,33% 16,67% 8,33% 8,33% 25,00% 8,33%
	Violazione di Reliability per il requisito SR3.	Log Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering Secure logging and auditing Trusted third parties Log Data Separation	3 3 6 3 3 3 3 3 3	5,56% 5,56% 22,22% 5,56% 11,11% 38,89% 5,56% 5,56% 5,56%
	Violazione di Resilience per il requisito SR3.	Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering Secure logging and auditing Trusted third parties Log Data Separation Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering Secure logging and auditing Trusted third parties Log Data Separation Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering Secure logging and auditing Trusted third parties Log Data Separation	3 3 3 6 3 3 3 3 3 3 6 21 54 3 3 3 3 3 3 3 3 3 3 6	33,33% 33,33% 33,33% 33,33% 33,33% 38,89% 5,56% 5,56% 5,56% 5,56% 11,11% 25,00% 8,33% 16,67% 8,33% 22,22% 5,56% 11,11% 38,89% 5,56% 5,56% 5,56% 5,56% 66,67%

Figura 43: Prioritize Requirements pt.1

2.5 Risk Reduction

REQUISITO	MISURA DI CONTROLLO	VALORE RISCHIO RESIDUO TOTALE	RISCHIO RESIDUO MAX	RISCHIO %
-----------	---------------------	-------------------------------	---------------------	-----------

SR4	Violazione di Authenticity per il requisito SR4.	Log Strong Authentication ACLs Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering Secure logging and auditing	3 3 12 3 12 9 6 6	8,33% 8,33% 33,33% 8,33% 33,33% 25,00% 16,67% 16,67%
	Violazione di Assurance per il requisito SR4.	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols Trusted third parties Throttle Quotas Tamper-resistant protocols Data Separation	9 3 18 6 21 3 3 9 6	14,29% 4,76% 28,57% 9,52% 33,33% 9,52% 4,76% 14,29% 9,52%
	Violazione di Accountability per il requisito SR4.	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols Filtering Trusted third parties	6 2 8 4 6 2 2	16,67% 5,56% 22,22% 11,11% 16,67% 5,56% 5,56%
	Violazione di Safety per il requisito SR4.	Log Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering ACLs Data Separation	3 3 12 9 3 6 6	8,33% 8,33% 33,33% 25,00% 8,33% 16,67% 16,67%
	Violazione di Reliability per il requisito SR4.	Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering Secure logging and auditing Log Trusted third parties Data Separation	3 21 9 6 15 6 3 6	5,56% 38,89% 16,67% 11,11% 27,78% 11,11% 11,11% 11,11%
	Violazione di Resilience per il requisito SR4.	Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering ACLs	3 6 9 3 3	33,33% 66,67% 100,00% 33,33% 33,33%
	Violazione di Authenticity per il requisito SRS.	Log Strong Authentication ACLs Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering Secure logging and auditing	3 3 12 3 12 9 6 6	8,33% 8,33% 33,33% 8,33% 33,33% 25,00% 8,33% 16,67%
	Violazione di Assurance per il requisito SRS.	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols Filtering Trusted third parties Throttle Quotas Tamper-resistant protocols Data Separation	9 3 18 6 21 6 3 3 9 6	14,29% 4,76% 28,57% 9,52% 33,33% 9,52% 4,76% 14,29% 9,52% 11,11%
	Violazione di Accountability per il requisito SRS.	Log Strong Authentication ACLs Secure logging and auditing Authentication protocols Filtering Trusted third parties	6 2 8 4 6 2 2	16,67% 5,56% 22,22% 11,11% 16,67% 5,56% 5,56%
	Violazione di Safety per il requisito SRS.	Log Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering ACLs Data Separation	3 3 12 9 3 6 6	8,33% 8,33% 33,33% 25,00% 8,33% 16,67% 16,67%
	Violazione di Reliability per il requisito SRS.	Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering Secure logging and auditing Log Trusted third parties Data Separation	3 21 9 6 15 6 3 6	5,56% 38,89% 16,67% 11,11% 27,78% 11,11% 11,11% 11,11%
	Violazione di Resilience per il requisito SRS.	Throttle Quotas Authentication protocols Tamper-resistant protocols Filtering ACLs	3 6 9 3 3	33,33% 66,67% 100,00% 33,33% 33,33%

Figura 44: Prioritize Requirements pt.2

2.5 Risk Reduction

REQUISITO	MISURA DI CONTROLLO	VALORE RISCHIO RESIDUO TOTALE	RISCHIO RESIDUO MAX	RISCHIO %
Violazione di Authenticity per il requisito SR6.	Log	3		6,67%
	Strong Authentication	3		6,67%
	ACLs	18		40,00%
	Throttle Quotas	3	45	6,67%
	Authentication protocols	3		6,67%
	Tamper-resistant protocols	6		13,33%
	Filtering	6		13,33%
	Message authentication codes	3		6,67%
	Log	3		5,56%
	Strong Authentication	6		11,11%
Violazione di Assurance per il requisito SR6.	ACLs	21		38,89%
	Throttle Quotas	3	54	5,56%
	Authentication protocols	9		16,67%
	Tamper-resistant protocols	3		5,56%
	Filtering	3		5,56%
	Run applications with least privilege	3		5,56%
	Log	3		11,11%
	Strong Authentication	6	27	22,22%
	ACLs	12		44,44%
Violazione di Accountability per il requisito SR6.	Authentication protocols	3		11,11%
	Log	3		6,67%
	Throttle Quotas	3		6,67%
	Authentication protocols	9	45	20,00%
	Tamper-resistant protocols	3		6,67%
	Filtering	3		6,67%
	ACLs	18		40,00%
	Strong Authentication	3		6,67%
Violazione di Safety per il requisito SR6.	Throttle Quotas	3		8,33%
	Authentication protocols	9	36	25,00%
	Tamper-resistant protocols	3		8,33%
	Filtering	6		16,67%
	ACLs	21		58,33%
	Strong Authentication	3		8,33%
	Throttle Quotas	3		33,33%
Violazione di Reliability per il requisito SR6.	Authentication protocols	3	9	33,33%
	Tamper-resistant protocols	3		33,33%
	Filtering	3		33,33%
	ACLs	6		66,67%
	Secure logging and auditing	3		8,33%
	Log	6		9,52%
Violazione di Authenticity per il requisito SR7.	Strong Authentication	3		4,76%
	ACLs	18	36	50,00%
	Throttle Quotas	3		8,33%
	Authentication protocols	6		16,67%
	Tamper-resistant protocols	3		8,33%
	Filtering	6		16,67%
Violazione di Assurance per il requisito SR7.	Secure logging and auditing	3		8,33%
	Log	6	63	19,05%
	Strong Authentication	3		9,52%
	ACLs	24		38,10%
	Secure logging and auditing	3		4,76%
	Authentication protocol	12		19,05%
Violazione di Accountability per il requisito SR7.	Filtering	6	36	9,52%
	Trusted third parties	3		4,76%
	Throttle Quotas	3		4,76%
	Tamper-resistant protocols	3		4,76%
	Data Separation	3		4,76%
	Log	6		16,67%
Violazione di Safety per il requisito SR7.	Strong Authentication	3	36	8,33%
	ACLs	15		41,67%
	Secure logging and auditing	3		8,33%
	Authentication protocol	6		16,67%
	Filtering	3		8,33%
	Trusted third parties	3		8,33%
Violazione di Reliability per il requisito SR7.	Log	2	54	5,56%
	Throttle Quotas	2		5,56%
	Authentication protocols	4		11,11%
	Tamper-resistant protocols	2	36	5,56%
	Filtering	2		5,56%
	ACLs	6		16,67%
Violazione di Resilience per il requisito SR7.	Data Separation	2		5,56%
	Throttle Quotas	2		3,70%
	Authentication protocols	8		14,81%
	Tamper-resistant protocols	2		3,70%
	Filtering	4		7,41%
	ACLs	14		25,93%
Violazione di Resilience per il requisito SR7.	Secure logging and auditing	2	9	3,70%
	Log	2		3,70%
	Trusted third parties	2		3,70%
	Data Separation	2		3,70%
	Throttle Quotas	3		33,33%
	Authentication protocols	3		33,33%
Violazione di Resilience per il requisito SR7.	Tamper-resistant protocols	3	6	33,33%
	Filtering	3		33,33%
	ACLs	6		66,67%

Figura 45: Prioritize Requirements pt.3

2.5 Risk Reduction

REQUISITO	MISURA DI CONTROLLO	VALORE RISCHIO RESIDUO TOTALE	RISCHIO RESIDUO MAX	RISCHIO %
Violazione di Authenticity per il requisito SR8.	Log	2		5,56%
	Strong Authentication	2		5,56%
	ACLs	12		33,33%
	Throttle Quotas	2	36	5,56%
	Authentication protocols	4		11,11%
	Tamper-resistant protocols	2		5,56%
	Filtering	4		11,11%
	Secure logging and auditing	2		5,56%
	Data Separation	2		5,56%
	Log	6		11,11%
Violazione di Assurance per il requisito SR8.	Strong Authentication	3		5,56%
	ACLs	21		38,89%
	Secure logging and auditing	3		5,56%
	Authentication protocols	9	54	16,67%
	Filtering	6		11,11%
	Trusted third parties	3		5,56%
	Throttle Quotas	3		5,56%
	Tamper-resistant protocols	3		5,56%
	Data Separation	3		5,56%
	Log	4		11,11%
Violazione di Accountability per il requisito SR8.	Strong Authentication	2		5,56%
	ACLs	10		27,78%
	Secure logging and auditing	2	36	5,56%
	Authentication protocols	4		11,11%
	Filtering	2		5,56%
	Trusted third parties	2		5,56%
	Data Separation	1		5,56%
	Throttle Quotas	1		3,70%
	Authentication protocols	3		6,67%
	Tamper-resistant protocols	1		2,22%
Violazione di Safety per il requisito SR8.	Filtering	2		4,44%
	ACLs	6	45	13,33%
	Secure logging and auditing	1		2,22%
	Log	1		2,22%
	Trusted third parties	1		2,22%
	Data Separation	1		2,22%
	Throttle Quotas	3		33,33%
	Authentication protocols	3		33,33%
	Tamper-resistant protocols	3	9	33,33%
	Filtering	3		33,33%
Violazione di Reliability per il requisito SR8.	ACLs	6		66,67%

Figura 46: Prioritize Requirements pt.4

2.5 Risk Reduction

Security Specification Definition Nella tabella della Fig.47 vengono messi in relazione i risultati dei *value-to-cost ratios* e delle percentuali dei rischi residui calcolati precedentemente.

Da tale confronto si ottiene un indice chiamato *Value-to-Risk* calcolato attraverso un'apposita matrice. In base ai risultati ottenuti dalla matrice, sono state assegnate delle etichette a ciascuna misura di controllo; questo aiuta nel valutare quali di esse implementare.

La scala cromatica utilizzata indica quali misure sono "migliori" e quali peggiori secondo tale confronto. Sono state escluse le misure di controllo con colore rosse e giallo; sono state invece considerate, per la prossima analisi, quelle con colore verde chiaro e verde scuro.

Nella Fig. 48 vengono riportate soltanto le misure di controllo rimanenti dopo tale analisi.

2.5 Risk Reduction

REQUISITO	MISURA DI CONTROLLO	VALORE COSTO TOTALE *	VALUE-TO-COST RATIO	RISCHIO %	Value-to-risk
SR1	ACLs	1	3	22,52%	
	Filtering	2	1,5	37,04%	
	Mirroring	3	1	7,41%	
	Encryption	2	1,5	33,33%	
	Data Separation	3	1	22,22%	
SR2	ACLs	1	3	44,44%	
	Filtering	2	1,5	27,78%	
	Mirroring	3	1	5,56%	
	Encryption	2	1,5	8,33%	
	Data Separation	3	1	16,67%	
SR3	Tamper-resistant protocols	3	1	5,56%	
	Message authentication codes	1	3	5,56%	
	Log	2	1,5	11,11%	
	Strong Authentication	1	3	5,56%	
	ACLs	1	3	66,67%	
SR4	Throttle Quotas	3	1	33,33%	
	Authentication protocols	3	1	33,33%	
	Tamper-resistant protocols	3	1	33,33%	
	Filtering	2	1,5	33,33%	
	Secure logging and auditing	3	1	5,56%	
SR5	Trusted third parties	1	3	5,56%	
	Data Separation	3	1	8,33%	
	Log	2	1,5	16,67%	
	Strong Authentication	1	3	5,56%	
	ACLs	1	2	33,33%	
SR6	Throttle Quotas	3	0,67	33,33%	
	Authentication protocols	3	0,67	66,67%	
	Tamper-resistant protocols	3	0,67	100,00%	
	Filtering	2	1	33,33%	
	Secure logging and auditing	3	0,67	16,67%	
SR7	Trusted third parties	1	2	5,56%	
	Data Separation	3	0,67	16,67%	
	Log	2	1,5	11,11%	
	Strong Authentication	1	3	22,22%	
	ACLs	1	3	66,67%	
SR8	Throttle Quotas	3	1	33,33%	
	Authentication protocols	3	1	33,33%	
	Tamper-resistant protocols	3	1	33,33%	
	Filtering	2	1,5	33,33%	
	Secure logging and auditing	3	1	8,33%	
	Trusted third parties	1	3	8,33%	
	Data Separation	3	1	5,56%	
	Log	2	1	11,11%	
	Strong Authentication	1	2	5,56%	
	ACLs	1	2	66,67%	
	Throttle Quotas	3	0,67	33,33%	
	Authentication protocols	3	0,67	33,33%	
	Tamper-resistant protocols	3	0,67	33,33%	
	Filtering	2	1	33,33%	
	Secure logging and auditing	3	0,67	5,56%	
	Trusted third parties	1	2	5,56%	
	Data Separation	3	0,67	5,56%	

Value-to-cost	Risk		
	High	Medium	Low
Value-to-cost	High	Medium	Low

NOTA: implementiamo soltanto le misure di controllo verde e verde mare.

Figura 47: Valutazione Misure di Controllo

L'ultimo passo consiste nel confrontare il rischio residuo associato a ciascuna tecnica di controllo (fra le rimanenti), abbinata ad un determinato requisito di sicurezza, con il rischio inherente da cui si è partiti. La scelta effettiva ricade su quelle misure per cui il rischio residuo sia strettamente minore del rischio

2.5 Risk Reduction

	REQUISITO	ATTACCO	RISCHIO INIZIALE	MISURA DI CONTROLLO	COSTO RISCHIO INIZIALE	RISCHIO REDUITO
GK1	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	1	3
GK2	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK3	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3
GK4	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK5	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK6	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3
GK7	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK8	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK9	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3
GK10	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK11	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK12	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3
GK13	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK14	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK15	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3
GK16	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK17	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK18	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3
GK19	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK20	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK21	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3
GK22	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK23	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK24	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3
GK25	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK26	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK27	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3
GK28	Violazione di Confidentiality per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK29	Violazione di Integrity per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	3
		Memorizzazione del sensibile	6	AC1a	2	3
		Modifica	6	Encryption	1	3
		Eliminazione	6	AC1a	2	3
GK30	Violazione di Availability per il requisito SEL.	Accesso non autorizzato	6	AC1a	1	2
		Interazione	6	Encryption	2	3
		Consegnamento dell'evento	6	AC1a	1	2
		Modifica	6	Filtering	2	3

Figura 48: Security Requirements Definition

inerente, indicando di fatto un effettivo beneficio.

Nella Fig.49 vengono riportate le misure di controllo per ciascun attacco che è stato, a seguito della analisi effettuate, deciso di implementare.

2.5 Risk Reduction

Le misure di controllo evidenziate sono quelle che, confrontando rischio inerente e rischio residuo, considerando anche il costo complessivo, è stato deciso di non implementare.

Figura 49: Security Requirements Definition - Scelte Effettuate

3 Blockchain e Smart Contracts

3.1 Distributed Ledger Technology

Si supponga che due utenti vogliano fare una transazione con la possibilità di non doversi per forza fidare l'un dell'altro o fare affidamento su un'entità terza. Interpellando un numero elevato di persone come testimoni si abbassa la probabilità di alterazione del contratto, ovvero è improbabile che più testimoni si mettano d'accordo per inserire delle clausole fasulle che andrebbero a gravare su una delle due parti. Proprio da questo concetto nasce la tecnologia della Distributed Ledger, ovvero un "libro" contenente transazioni condiviso, replicato, sincronizzato e decentralizzato. Ogni testimone della rete, infatti detiene una copia dello stesso registro che deve essere aggiornato ogni volta viene registrata una nuova transazione dopo che è stata convalidata dalla rete di testimoni. Una DLT(Distributed Ledger Technology) si distingue dalle altre per tre aspetti fondamentali:

- Struttura dati;
- Protocollo utilizzato in fase di registrazione della transazione;
- Algoritmo del consenso che regola i meccanismi di accordo sulla convalida della transazione.

3.2 Blockchain

La Blockchain è una particolare tecnologia Distributed Ledger che fu implementata diversi anni fa per la registrazione di transazioni basate su criptovalute (come Bitcoin). L'aumento dei paesi che hanno legalizzato l'uso di questa tecnologia ha permesso di condurre delle ricerche più approfondite su di essa, che si è dimostrata utile anche in altre applicazioni oltre a quella delle criptovalute. Il principale vantaggio dell'uso di Blockchain, in alternativa a soluzioni più comuni e meno costose in termini di potenza computazionale e consumo energetico (come database e cloud storage), risiede nei requisiti di sicurezza che Blockchain riesce a garantire: autenticità, disponibilità, anonimato, responsabilità e integrità. Una rappresentazione del workflow di una blockchain è visibile nella Fig. 50

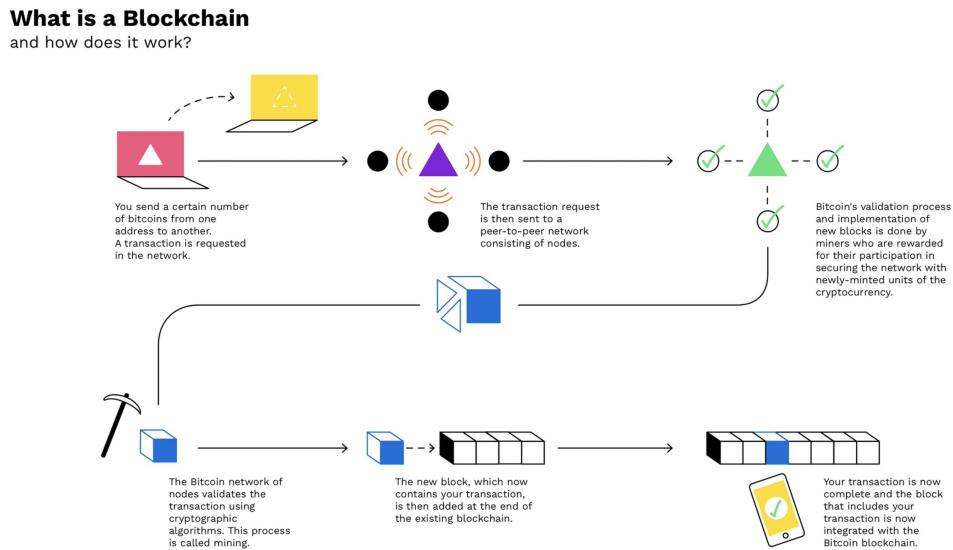


Figura 50: Workflow di una blockchain

Struttura Dati Come dice il nome stesso la Blockchain non è altro che una catena di blocchi all'interno del quale sono registrate delle transazioni. Ciascun blocco ha una capacità di storage limitata, inoltre è identificato da un timestamp che ne indica la data e l'orario di creazione e da un nonce che è invece un indicatore di freschezza. Ciascun blocco ha un puntatore al blocco successivo che è rappresentato da un hash del blocco precedente in modo tale che, a partire dalla testa della catena, sia possibile risalire al blocco iniziale (una logia in cui il primo elemento della lista è anche il primo ad essere stato scritto).

Protocollo di registrazione delle transazioni Ciascun utente può interagire con la Blockchain attraverso un wallet o portafoglio elettronico identificato da una stringa di caratteri generata a partire da una coppia di chiavi pubblico/privata in possesso dell'utente fisico (questo meccanismo permette all'interno della Blockchain di risalire all' ID del wallet garantendo la responsabilità ma non permette di identificare l'utente fisico, assicurando dunque anonimato). Supponendo che l'utente A voglia inviare una somma di denaro all'utente B, cioè voglia eseguire una transazione economica, la rete di "testimoni" che in Blockchain prendono il nome di nodi validatori o più semplicemente minatori, deve convalidare la transazione. In questo caso la convalida consiste nel risalire la Blockchain fino al blocco iniziale per analizzare tutte le transazioni validate da e verso A per stabilire dunque un saldo del wallet. Se il saldo supera la somma richiesta dalla transazione allora questa è convalidata, ma i dati non sono subito resi consistenti in quanto i nodi attendono che si accumulino un certo numero di transazioni prima di costruire un nuovo blocco da convalidare e aggiungere alla Blockchain. Si parla di consistenza eventuale, infatti un utente potrebbe paradossalmente andare a spendere dei soldi già spesi ma che non ancora sono stati scalati. Ovviamente Blockchain adotta una serie di meccanismi per risolvere questo problema di inconsistenza.

Algoritmo del consenso Gli algoritmi del consenso permettono di stabilire un metodo con il quale i nodi si accordano sulla validità di una transazione o di un blocco. Diverse sono le tipologie di algoritmi che meglio si adattano in base alla tipologia di Blockchain. Esistono infatti:

- **Blockchain pubbliche:** sono senza permessi e chiunque può entrare a far parte della comunità di nodi validatori. Essendo pubbliche richiedono algoritmi più sicuri in cui aumenta la soglia di consensi minimi ($50\% + 1$) per decretare la validità di una transazione. Questi si basano su delle sfide crittografiche o scommesse che decretano un vincitore che diventa di diritto nodo principale per quella transazione. Per incentivare la partecipazione dei nodi alla sfida e conseguente convalida dei blocchi, si utilizzano meccanismi di ricompensa in criptovalute come Bitcoin. Sono indubbiamente gli algoritmi più dispendiosi in termini di potenza computazionale e quindi energia con tempi di convalida che possono andare dai 10 minuti all'ora di tempo. Sono un esempio gli algoritmi Proof of Work e Proof of Stake.
- **Blockchain private:** la Blockchain è di proprietà di un determinato ente che distribuisce a sua discrezione la possibilità di entrare nella comunità dei nodi. In questo caso si possono usare algoritmi meno sicuri e quindi con una soglia più bassa di consensi minimi (33%). Inoltre non vengono usati dei meccanismi di ricompensa per premiare i nodi validatori anche se le transazioni passano sempre prima sotto il vaglio di un nodo principale che si preoccupa di inviare queste in multicast verso i nodi secondari che si occuperanno della convalida attraverso scambio di messaggi. Ovviamente i consumi sono inferiori non essendovi delle sfide crittografiche. Un esempio è rappresentato dall'algoritmo Tolleranze ai guasti di tipo Bizantino.
- **Blockchain di consorzio di enti:** sono come le Blockchain private, ma in questo caso sono più enti organizzativi ad essere possessori della Blockchain (enti governativi, nazioni, aziende).

Ogni transazione crittografata e i relativi dati vengono inseriti in un blocco, sottoposti alla verifica e all'approvazione dei partecipanti. I "Miners" (minatori) competono alla soluzione di una sfida crittografica tramite degli algoritmi di consenso distribuito, alcuni dei quali sono:

- Proof of work: il minatore che valida il blocco è il primo a risolvere la sfida crittografica la quale, visto l'elevato consumo energetico, è in mano ai pochi attori che hanno grandi capacità di calcolo. Questo è scongiurato grazie anche alla nascita di unioni di minatori che condividono la risorsa aumentando così le probabilità di vittoria.
- Proof of stake: il minatore che valida il blocco è scelto sulla base della quantità di valuta posseduta. Per evitare che vincano sempre gli stessi, chi vince una sfida non può partecipare ad un certo numero di sfide successive.

3.3 Smart Contracts

Col tempo diverse Blockchain hanno iniziato a permettere l'uso di Smart Contracts, letteralmente contratti intelligenti. Questi sono scritti sotto forma di programma ed eseguiti in un determinato linguaggio,

3.4 Ethereum

in cui due o più parti si accordano su una o più clausole contrattuali.

Il vantaggio principale di questa tecnologia è che l'esecuzione del contratto non è più ambigua ed avviene in maniera automatica rispetto ad un normale contratto. Questa tecnologia può essere usata solo nei paesi che ne garantiscono l'effettiva valenza giuridica.

Così come gli utenti, anche gli Smart Contracts possono interfacciarsi con la Blockchain attraverso l'uso di un portafoglio elettronico. Una volta che il contratto è stato sottoscritto è necessaria la firma di tutte le parti che vi entrano in gioco prima che lo stesso possa essere eseguito. Una volta eseguito, si mette in attesa di una transazione in ingresso, che non appena arrivata viene elaborata dal contratto che genera una transazione in output rimettendosi in attesa di un'altra transazione in input.

L'obiettivo finale degli Smart Contracts è quello di assicurare una sicurezza superiore ai normali contratti oggi utilizzati e ridurre i costi di transazione associati alla contrattazione.

Analizzate queste due tecnologie il passo successivo sarà confrontarne le caratteristiche e i potenziali vantaggi in modo tale da poter sottolineare il riscontro pratico all'interno del sistema da implementare.

Gli Smart Contracts possono essere utilizzati per sviluppare le dapps (Decentralized Applications).

Uno smart contract è scritto in una transazione, pertanto è tracciabile ed irreversibile; le sue clausole possono essere auto-ottemperanti in quanto il contratto è un algoritmo e quindi possiede un'unica interpretazione (diversamente da quello che accade con i comuni contratti scritti in linguaggio naturale e potenzialmente soggetti ad interpretazioni). Con gli smart contracts si vogliono raggiungere livelli di sicurezza maggiori rispetto a quelli dei contratti tradizionali e ridurre i costi di transazione. La natura delle blockchain le rende particolarmente adatte a questo nuovo tipo di contratti e anche in questo caso si applica il proof of work.

Ethereum utilizza gli smart contract come algoritmi eseguibili scritti in linguaggio Solidity per e utilizza Ether come criptovaluta per ricompensare i minatori e pagare l'esecuzione di codice.

3.4 Ethereum

Ethereum è una piattaforma di distributed computing pubblica, open source e basata su blockchain. Più nello specifico è una generalizzazione (general purpose) della blockchain Bitcoin in cui si possono scrivere dati di diversa natura, non per forza valute, ma anche algoritmi eseguibili che rimangono memorizzati nella blockchain: i dati di transazione possono quindi essere sostituiti con dati relativi a qualsiasi altro servizio. La **criptovaluta** è sostituita da un **token** utilizzato per accedere al servizio stesso. La possibilità di avere la blockchain general purpose è anche dovuta al fatto che la rete Ethereum ingloba il concetto di smart contract, ovvero un programma informatico che consente di facilitare, verificare o imporre la negoziazione e l'esecuzione di un contratto senza la necessità di terze parti.

4 Design

4.1 Secure Design

Risk Driver Design Questa corrisponde alla fase di progettazione software vera e propria. In particolare, si prendono tutti i requisiti sia funzionali che non, prodotti dalle sezioni precedenti e si cerca di tradurli in componenti ed architetture software. Nella scelta delle componenti è necessario attuare un compromesso tra prestazioni e costi. Si ragiona su architetture di protezione che permettono di ridurre le vulnerabilità e su architetture di ridondanza, diversità e distribuzione che invece riducono l'impatto. Nel fare ciò ancora una volta bisogna trovare dei compromessi in modo tale da poter gestire eventuali conflitti (ad esempio tra ridondanza e architetture di protezione, che dovendo essere implementate su ciascun sistema ridondante, aumenterebbero di gran lunga i costi di progettazione).

In letteratura esiste molta documentazione relativa alla progettazione del software sicuro. In questo caso è stato fatto riferimento alle linee guida emesse da OWASP (Open Web Application Security Project) e da Sommerville.

Sulla base dei concetti espressi da queste linee guida si cerca di contestualizzare l'utilizzo della Blockchain come architettura al problema preso in esame. Per fare ciò si parte dalla tabella riassuntiva delle tecniche di controllo utilizzate precedentemente riportata (Fig. 49). Da questa si evince che le più frequenti tecniche di controllo adottate sono le seguenti:

1. ACLs;
2. Filtering;
3. Log

Le principali tecnologie che implementano queste tecniche di controllo sono: Blockchain, Web Server con DB e DB locale. La scelta è ricaduta sulla tecnologia Blockchain come da specifica progettuale. Riportiamo quindi brevemente le motivazioni a supporto di tale specifica attraverso l'argomentazione delle principali linee guida emesse da OWASP e Sommerville:

- **Stabilire dei default sicuri:** imponendo delle impostazioni sicure di default, risulterebbe difficile per un utente finale modificare le impostazioni di sicurezza, inoltre, la tecnologia Blockchain potrebbe essere protetta da un front-end che impedirebbe l'accesso alle impostazioni all'utente finale;
- **Privilegi minimi:** la tecnologia Blockchain permette di creare account con determinati privilegi, inoltre un'eventuale front-end potrebbe bloccare l'accesso a livello superiore;
- **Difesa in profondità:** la tecnologia Blockchain implementa meccanismi di difesa aggiuntivi come funzioni hash, ovvero primitive crittografiche in grado di aumentare la sicurezza generale del sistema;
- **Fallire in maniera sicura:** i meccanismi che Blockchain utilizza per gestire le transazioni fanno sì che in caso di fallimento la transazione non viene eseguita senza quindi mandare in blocco il sistema;
- **Sicurezza per oscurità da evitare:** essendo Blockchain una tecnologia open design, ci sarà tutta una comunità che studia la struttura per verificare la presenza di errori, che poi possono essere corretti;
- **Evitare i single point of failure:** l'utilizzo di più nodi sul quale è salvata la Blockchain abbassa il rischio di single point of failure;
- **Utilizzare il log:** la tecnologia Blockchain offre già la possibilità di tenere traccia di tutto ciò che avviene al suo interno.
- **Ridondanza e diversità:** la tecnologia fa leva proprio sulla distribuzione delle informazioni su diversi nodi, garantendo la ridondanza e la diversità;
- **Specificare sempre il formato degli input:** la Blockchain dotata di front-end permette la sanificazione degli input;

4.1 Secure Design

- **Possibilità di ripristinare il funzionamento del sistema:** grazie all'uso dei log è possibile ripristinare uno stato precedente in seguito ad un malfunzionamento.

5 Tecnologie Utilizzate

Il progetto qui sviluppato, si pone come obiettivo la realizzazione di una Web App, che rispetti i requisiti e le specifiche qui sopra citate. Riportiamo quindi una breve sintesi delle tecnologie utilizzate al fine della realizzazione della suddetta applicazione.

5.1 Quorum

La blockchain adotta nella fase di implementazione è Quorum, un fork della blockchain Ethereum che ne implementa una sua versione privata (permissioned invece che permissionless). Presentando quindi alcuni miglioramenti, in particolare possiede le seguenti caratteristiche:

- Supporto per transazioni private;
- Meccanismi di consenso basati su voto multiplo;
- Gestione dei permessi di rete;
- Performance migliori.



Figura 51: Logo Quorum

Queste caratteristiche hanno guidato la nostra decisione di scegliere Quorum come blockchain da adottare per il progetto.

5.2 Solidity

Per lo sviluppo degli smart contract si è fatto uso di **Solidity**, un linguaggio orientato agli oggetti e ad alto livello, realizzato appositamente per questo scopo: per blockchain che fanno uso della Ethereum Virtual Machine (EVM). Una volta realizzati, tali smart contract vengono compilati (tramite un compilatore come solc) in un oggetto che contiene il codice ABI, rappresentante lo schema del contratto, e il bytecode, che verrà eseguito all'interno della EVM.



Figura 52: Solidity Logo

Per sviluppare i contratti in maniera agevole abbiamo fatto utilizzo dell'IDE **Remix**, reso disponibile dagli autori di Ethereum. Questo IDE ci ha consentito di sviluppare in maniera agile ed efficace i contratti, mettendo a disposizione diversi strumenti per la loro compilazione ed il loro deploy all'interno di una rete temporanea, che consente un testing rapido direttamente sul browser.

5.3 Truffle

Dopo aver sviluppato i contratti ci siamo occupati dell’interfacciamento tra i contratti e la nostra blockchain. Per far ciò, abbiamo utilizzato **Truffle**.

Truffle è un ambiente di sviluppo che permette di compilare, effettuare il linking, distribuire e gestire i binari degli smart contract facilmente. Si interfaccia con Ethereum e Quorum ed è facilmente utilizzabile poiché è sufficiente installarlo tramite il package manager npm. Per effettuare il deploy sulla blockchain è sufficiente realizzare dei file javascript, che permettono di effettuare il caricamento dinamico degli smart contract all’interno della blockchain. Nel caso del progetto, Truffle è stato utilizzato per effettuare rapidamente e comodamente il deploy degli smart contract all’interno della blockchain in fase di testing.



Figura 53: Truffle Logo

5.4 Node.js

Node.js è un runtime environment open source e multiplattforma che consente di realizzare applicazioni lato server in JavaScript. Tale runtime è destinato ad essere eseguito direttamente su un computer e non sul browser.

I vantaggi di questo ambiente di sviluppo sono numerosi: ha delle buone performance, è stato progettato per ottimizzare il throughput e la scalabilità nelle applicazioni web e permette l’utilizzo del **Node Package Manager (NPM)** che fornisce l’accesso a centinaia di migliaia di pacchetti riutilizzabili. Questi motivi, assieme alla popolarità dell’ambiente per quanto riguarda lo sviluppo di applicazioni distribuite su blockchain (D-Apps), hanno determinato la scelta di adottare Node.js come tecnologia per lo sviluppo dell’applicazione.

5.5 Express

Nello sviluppo della Web App si è deciso di utilizzare il framework **Express**. Express è un framework web veloce, non categorico e minimalista per Node.js, che fornisce meccanismi per gestire richieste HTTP in base ai diversi URL (rotte). Si integra facilmente con motori di rendering come **ejs5**, da noi utilizzato per realizzare le diverse pagine dell’applicazione. Inoltre, esistono numerose librerie per lavorare con cookie, sessioni, accessi utente, intestazioni di sicurezza e molto altro, che si integrano facilmente con il framework.

5.6 Web3

Per collegare l’applicazione alla blockchain, realizzando le chiamate e le transazioni agli smart contract presenti al suo interno, abbiamo fatto uso della libreria **web3.js**. Web3.js è un’API JavaScript che consente di interagire con un nodo Ethereum locale o remoto utilizzando HTTP, IPC o WebSocket. È la libreria più utilizzata in questo contesto, e presenta di una grande community che ne fa utilizzo.

5.7 MySQL

Al fine di una memorizzazione aggiuntiva di informazioni come le credenziali di accesso alla Web App, abbiamo dotato quest’ultima di un database. Il database oltre a far fronte allo storage delle credenziali, manterrà informazioni relative ai contratti presenti nella blockchain. In seguito a svariate analisi, abbiamo deciso di addottare MySQL. MySQL è un DBMS⁶, relazionale. Per una corretta interazione abbiamo utilizzato il pacchetto Sequelize che ci ha permesso le interazioni con il nostro database.

⁶DBSM \vdash Database Management System

6 Misure di sicurezza implementate

6.1 Interfacciamento dell'applicazione con la *blockchain*

Per quanto riguarda l'uso della blockchain all'interno dell'applicazione, abbiamo fatto innanzitutto uso dello strumento quorum-wizard⁷ per costruirla in locale. Considerate le capacità limitate dei calcolatori su cui è avvenuto lo sviluppo dell'applicazione, abbiamo scelto di adottare una blockchain composta soltanto da 3 nodi, in modo da limitare l'utilizzo della memoria RAM in fase di sviluppo.

Per quanto riguarda gli utenti, l'indirizzo dell'account della blockchain a cui è collegato viene memorizzato nel database MySQL assieme alle sue credenziali. Una volta effettuato il login, questo indirizzo verrà poi utilizzato per effettuare le chiamate ed eseguire le transazioni all'interno della blockchain.

Gli smart contract realizzati e implementati nella blockchain sono due. In particolare:

- Uno smart contract serve per la memorizzazione delle informazioni sugli eventi. Verranno creati e memorizzati tanti contratti quanti sono gli eventi, cioè ciascun contratto contiene le informazioni di un singolo evento. Al suo interno vengono riportate le seguenti informazioni:
 - L'*id* dell'evento;
 - Il titolo dell'evento;
 - Il luogo dell'evento;
 - La data dell'evento;
 - L'orario dell'evento;
 - L'artista che terrà l'evento;
 - Un timestamp;
 - La capienza dell'evento;
 - Lo stato dell'evento (attivo, annullato, concluso).
- Uno smart contract è riservato ai biglietti di uno specifico evento. Esso memorizza una lista con tutte le informazioni di tutti i biglietti venduti per uno specifico evento. In particolare, per ogni biglietto vengono memorizzati:
 - L'*id* del biglietto;
 - La tipologia del biglietto (Standard, Gold, Platinum);
 - Il timestamp del biglietto;
 - Il prezzo del biglietto;
 - Lo stato del biglietto (valido, annullato, invalidato);
 - L'indirizzo del wallet dell'utente che ha acquistato il biglietto.

⁷<https://github.com/jpmorganchase/quorum-wizard>

6.2 Login alla webapp

6.2 Login alla webapp

Il login alla webapp avviene sfruttando un sistema basato sulle sessioni, le informazioni sono quindi salvate e viaggiano all'interno di cookie. Questi cookie sono cifrati mediante una chiave da noi specificata e salvata localmente.

Le password degli utenti registrati alla webapp vengono memorizzate sul DB dopo averle cifrate utilizzando un algoritmo di *hashing*, questo per evitare di avere memorizzate le password in chiaro nel DB e di non permettere, in caso di accesso non autorizzato al DB, l'ottenimento di informazioni che permettono di effettuare il login alla webapp. Per effettuare l'*hashing* delle password si è utilizzato il pacchetto *bcrypt*⁸, che permette di effettuare l'*hashing* di una stringa in chiaro specificando anche un *salt rounds* così da aumentare l'efficacia dell'*hashing* e rendere più difficili attacchi di tipo *brute-force*.

6.3 Prevenzione per Cross-Site Request Forgery (CSRF)

La webapp contiene delle form per l'inserimento dati da parte degli utenti, potrebbero essere soggette quindi ad attacchi di tipo Cross-Site Request Forgery (CSRF). Tale attacco consiste in un'invocazione, da parte dell'attaccante, di un comando sfruttando un utente correttamente autenticato alla webapp a sua insaputa. Per evitare ciò alla generazione di una form viene generato anche un token che servirà poi per verificare l'effettiva validità della richiesta inviata da parte degli utenti, se il token inviato assieme alla richiesta corrisponde a quello generato assieme alla form la richiesta viene considerata valida, altrimenti no.

Per fare uso di questi token associati alle form si è utilizzato il pacchetto *csurf*⁹.

6.4 Funzione di logging

Per il monitoraggio dell'applicazione e delle azioni eseguite dagli utenti si è fatto uso di un sistema di monitoraggio, il quale utilizza dei file di log in cui memorizzare le azioni compiute dagli utenti e le richieste inviate alla webapp.

Si sono utilizzati i pacchetti *winston*¹⁰ e *morgan*¹¹. Il primo ha permesso di effettuare il monitoraggio delle azioni eseguite dagli utenti, mentre il secondo il monitoraggio delle richieste HTTP inviate alla webapp.

Questo meccanismo di *logging* permette di monitorare il funzionamento della webapp, individuando e correggendo rapidamente eventuali problemi sul funzionamento della stessa.

⁸<https://www.npmjs.com/package/bcrypt>

⁹<https://www.npmjs.com/package/csrf>

¹⁰<https://www.npmjs.com/package/winston>

¹¹<https://www.npmjs.com/package/morgan>

7 Guida Web App

7.1 Manuale Utente

La pagina di benvenuto accessibile tramite il link: localhost:3000, presenta la form di login, che attraverso le credenziali univoche per ciascun utente, permette l'accesso all'applicazione.

The screenshot shows a web browser window with the title bar "localhost". The main content area displays a "Login" form. At the top of the form is a "Username" field with the placeholder "Username". Below it is a "Password" field with the placeholder "Password". A blue "Submit" button is located below the password field. At the bottom of the form, there is a footer bar containing the text "Lorenzo Fratini - Emanuele Incicco - Federico Mischia - Andrea Pinciaroli - Denis Bernovschi".

Figura 54: Webapp Home page

Nel caso l'utente non sia ancora registrato è obbligatoria la sua registrazione, accessibile tramite il menu, all'interno della navbar e/o tramite il link: localhost:3000/user/new.

The screenshot shows a web browser window with the title bar "localhost". The main content area displays a registration form titled "Cybersecurity Project". The form fields include: "Nome" (Name) with placeholder "Inserisci nome"; "Cognome" (Surname) with placeholder "Inserisci cognome"; "Username" with placeholder "Enter username" and a note "We'll never share your information with anyone else."; "Password" with placeholder "Password"; and "Indirizzo Wallet" (Wallet Address) with placeholder "Inserisci account". A blue "Registrati" (Register) button is located at the bottom left of the form. At the bottom, there is a footer bar with the same authorship information as Figure 54.

Figura 55: Form New User

Una volta effettuata l'autenticazione, in base alla tipologia di utenza, si potranno effettuare determinate azioni, in particolare attraverso un meccanismo di ACLs abbiamo definito quelle che sono le azioni eseguibili da ciascuna categoria. Riportiamo quindi brevemente quelle che sono le categorie di utenza:

- Admin (Event Manager)
- Invalidator
- User (Ticket Buyer)

Ciascuna categoria possiede determinati permessi, che riassumiamo qui di seguito:

- Admin (Event Manager)

7.1 Manuale Utente

- Visualizzare la lista degli eventi;
- Visualizzare i dettagli di un evento;
- Acquistare un biglietto;
- Visualizzare la lista dei biglietti venduti;
- Concludere un evento;
- Annullare un evento;
- Creare un nuovo evento;
- Annullare un biglietto.

- Invalidator

- Visualizzare la lista degli eventi;
- Visualizzare i dettagli di un evento;
- Acquistare un biglietto;
- Visualizzare la lista dei biglietti venduti;
- Invalidare un biglietto.

- User (Ticket Buyer)

- Visualizzare la lista degli eventi;
- Visualizzare i dettagli di un evento;
- Acquistare un biglietto.

Effettuando il login ci si ritrova nella seguente pagina localhost:3000/user che riassume le informazioni sull'utente che si è appena autenticato.

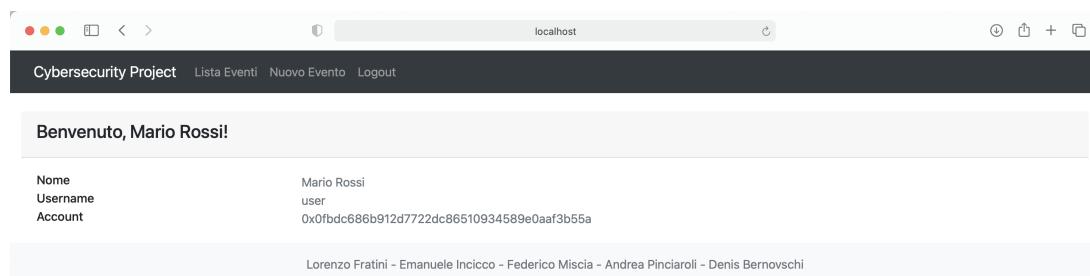


Figura 56: User Page

Come possiamo osservare nella navbar, compaiono alcune delle operazioni possibili. Scegliendo una delle operazioni se l'utente è autorizzato a compierla l'azione andrà a buon fine, in caso contrario comparirà un banner che informa l'utente che non è autorizzato a compiere quella operazione.

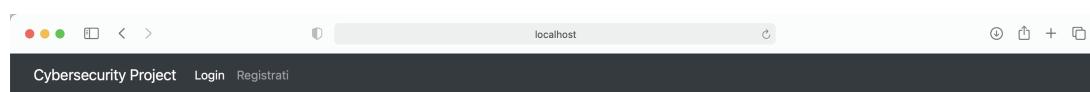


Figura 57: Navigation Bar

7.1 Manuale Utente

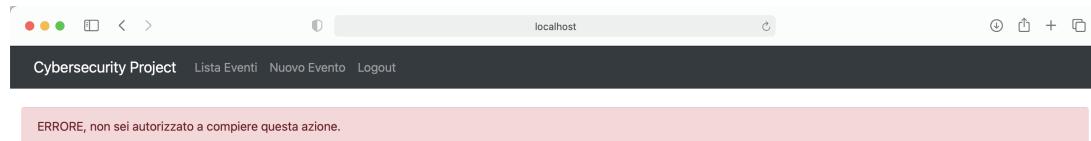


Figura 58: Error Navigation

Una possibile opzione è visionare la lista degli eventi, ove è possibile in primis acquistare un biglietto e/o visionare i dettagli relativi a ciascun evento.

The screenshot shows a browser window with a dark header bar containing the text "Cybersecurity Project" and navigation links. The main content is titled "Lista degli Eventi". It displays a table with two rows of event data:

ID	Evento	Titolo	Luogo	Data	Orario	Artista	Capienza	Stato	Timestamp
0							0/0	Attivo	2021-08-06T13:20:18.456Z
1	Negramaro Cover	Fano	21/09/2021	20:30	Negramaro	300/300	Attivo	2021-08-06T13:20:18.456Z	

Each row has three blue buttons at the end: "Dettagli evento", "Lista biglietti venduti", and "Acquista biglietto".

Figura 59: Lista degli Eventi

Come dicevamo poc'anzi, per visionare i dettagli di un evento sarà sufficiente premere su dettagli eventi e verremo reindirizzati alla pagina `localhost:3000/evento/id/*`.

The screenshot shows a browser window with a dark header bar containing the text "Cybersecurity Project" and navigation links. The main content is titled "Evento 'Negramaro Cover'". It displays the following details for the event:

- ID Evento: 1
- Luogo Evento: Fano
- Data Evento: 21/09/2021
- Orario Evento: 20:30
- Artista Evento: Negramaro
- Capienza Evento: 300/300
- Stato Evento: Attivo

At the bottom of the page are three blue buttons: "Acquista Biglietto", "Concludi Evento", and "Annulla Evento". A footer at the bottom of the page reads: "Lorenzo Fratini - Emanuele Incicco - Federico Mischia - Andrea Pinciaroli - Denis Bernovschki".

Figura 60: Dettagli Evento

Va notato che oltre all'opzione "acquista biglietto" sono presenti due ulteriori opzioni di carattere manageriale, per gestire l'evento, queste opzioni sono riservate all'Event Manager. Va osservato che l'opzione "Concludi Evento" va ad impostare l'evento come concluso in modo tale da garantire che nessuno possa acquistare biglietti relativi ad un evento concluso, analogo discorso va fatto per quanto concerne l'opzione "Annulla Evento", che permette all'Event Manager di annullare un evento.

7.1 Manuale Utente

Come si osserva nella Fig. 59 un’ulteriore funzione è quella relativa all’acquisto di un biglietto.



Figura 61: Acquisto Biglietto

L’utente, una volta selezionata l’opzione acquista biglietto, dovrà scegliere la tipologia del ticket che intende acquistare, e attendere la verifica del pagamento, la quale avviene off-chain. Nel caso in cui la verifica del pagamento ha esito negativo verrà visualizzato un banner che informa l’utente, il quale dovrà ripetere la procedura di acquisto del biglietto. In caso di esito negativo il biglietto non verrà registrato nella blockchain.



Figura 62: Errore Acquisto Biglietto

In caso contrario, se la verifica del pagamento restituisce un esito positivo, il biglietto verrà registrato e sarà visionabile nella lista dei biglietti venduti.

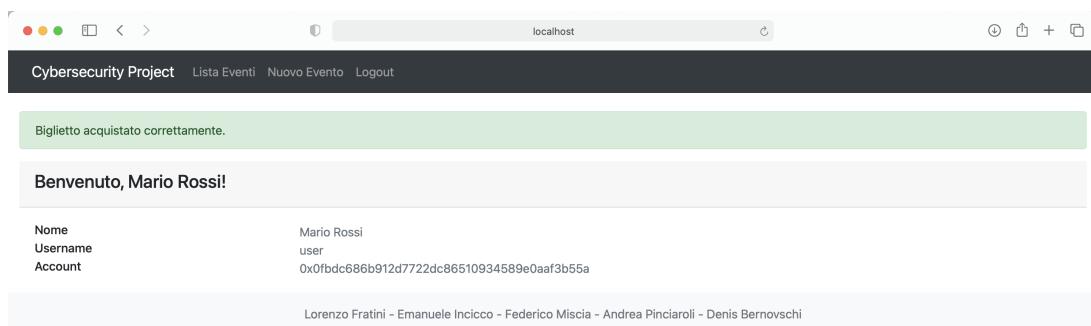


Figura 63: Acquisto Biglietto Corretto

7.1 Manuale Utente

La lista dei biglietti venduti è visibile a questo indirizzo localhost:3000/evento/id/*/biglietti, essa riporta una panoramica dei biglietti venduti per lo specifico evento selezionato.

La lista dei biglietti è accessibile da parte dell'Event Manager e dell'Invalidator, questo permette al primo di, eventualmente, annullare uno o più biglietti, questo soltanto nel caso in cui lo stato dell'evento corrispondente risulta anch'esso annullato, e al secondo di invalidare uno o più biglietti degli utenti che accedono all'evento, evitando così l'utilizzo del medesimo biglietto più volte.

ID Biglietto	Tipo Biglietto	Prezzo	Codice Sigillo	Stato	Timestamp	Indirizzo Account Intestatario		
0	Gold	39.99	UvkxxgglvdCFzkoF	Valido	2021-08-06T13:35:55.088Z	0x0FBdc686b912d7722dc86510934589E0AAf3b55A	Invalida Biglietto	Annulla Biglietto

Lorenzo Fratini - Emanuele Incicco - Federico Miscia - Andrea Pinciaroli - Denis Bernovschii

Figura 64: Lista Biglietti Venduti

Il bottone "Logout", infine, permette di effettuare la disconnessione dall'account.