



Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

Condividere informazioni in modo sicuro combinando Git e Blockchain

Laureando: Paolo Speziali
Relatore: Luca Grilli

Università degli Studi di Perugia - Dipartimento di Ingegneria
Corso di laurea triennale in Ingegneria Informatica ed Elettronica



A.D. 1308
unipg
DIPARTIMENTO
DI INGEGNERIA

A.A. 2020/2021



Indice

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

1 Il Problema

2 Concetti preliminari

3 L'Obiettivo

4 Il Software PineSU

5 Tecnologie utilizzate

6 Sviluppi futuri



La digitalizzazione

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

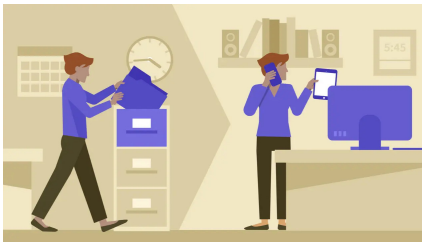
L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

È in atto, negli ultimi anni, un piano di **digitalizzazione** delle PA. Esso mira all'evoluzione tecnologica di tutte le sue mansioni e alla creazione di portali web per il cittadino. L'esigenza di questa trasformazione si è fatta sentire anche da parte dell'**Unione Europea**, che con il **Recovery Fund** ci sta fornendo i fondi per attuarla, ben **11,75 milioni di euro**.





Il problema della burocrazia

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

Il più grande avversario della digitalizzazione è la **burocrazia** italiana: i suoi processi sono **lenti** e **complessi** anche a causa dell'**importanza** dei documenti da gestire. È necessaria una **sburocratizzazione** grazie a degli strumenti digitali che permettano di **salvare**, **validare** e **condividere** documenti senza abbassare il **livello di sicurezza**.





Gli strumenti attuali

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

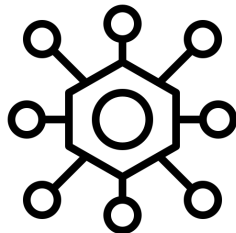
Uno strumento digitale solitamente segue uno di questi due paradigmi:

centralizzato e **distribuito**.

Nel primo un'entità centrale si occupa dell'**immagazzinamento** e della **verifica** dei dati degli utenti.

Ciò ha diversi **svantaggi**:

- Potenziali attacchi all'entità
- Possibile uso malevolo dei nostri dati
- Alti costi d'intermediazione



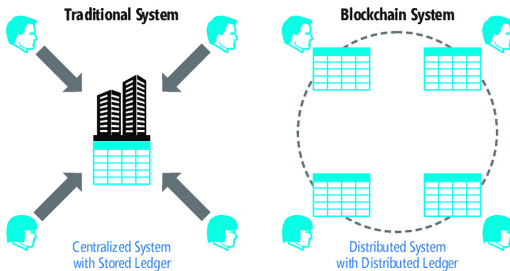


Strumenti distribuiti

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Usando invece **un'architettura distribuita**, sia per la **gestione dei file**, sia per la **verifica delle informazioni**, saremo in grado costruire uno strumento che può affidarsi alla parola di una **moltitudine di entità**, rendendo molto più complicati e rilevabili attacchi e manomissioni.





Funzioni crittografiche di hashing

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

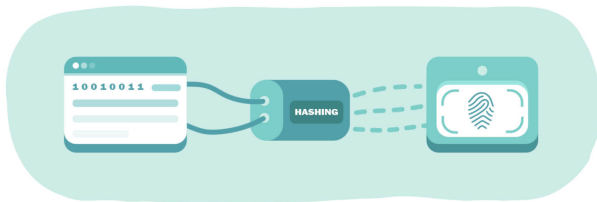
L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

Funzione che **associa**, a una qualsiasi sequenza m di lunghezza arbitraria in input, una sequenza in output $h(m)$ di lunghezza costante, seguendo alcune proprietà che la rendono *crittograficamente sicura*. Ciò impedisce di risalire all'input originale e facilita i **controlli di integrità sui file**.





Git

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

Git è il sistema di controllo di versione (**VCS**) distribuito più diffuso al mondo.

Esso agevola la gestione **distribuita** di insiemi di file e directory. Un VCS considera tali insiemi unità chiamate **repository**.

Git ci permette di:

- **Tracciare** le modifiche in una repository.
- **Ripristinare** le repository ad uno stato precedente.
- **Condividere** le repository con il loro storico dei cambiamenti.

e molto altro. . .





Blockchain

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

La **blockchain** è un registro in continua crescita di record chiamati **blocchi**, collegati l'uno all'altro come in una **catena** grazie a **metodi crittografici**. Essa è:

- **Immutabile.**
- **Distribuita.**
- **Estremamente sicura.**





Blockchain

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

La **blockchain** è un registro in continua crescita di record chiamati **blocchi**, collegati l'uno all'altro come in una **catena** grazie a **metodi crittografici**. Essa è:

- **Immutabile.**
- **Distribuita.**
- **Estremamente sicura.**



È alla base delle reti di criptovalute, come **Ethereum**, su cui si possono anche costruire applicazioni decentralizzate con gli **Smart Contract**.



Perché blockchain?

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

L'utilizzo della blockchain nel progetto è giustificato da:

- **Immutabilità** → Garantisce integrità dei dati.
- **Decentralizzazione** → Resistenza al Single Point Of Failure.
- **Disintermediazione** → Eliminazione di *middle-men* e dei loro costi.
- **Validazione peer-to-peer** → Potere distribuito.



Il problema della blockchain

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

Vogliamo usare la blockchain per **immagazzinare** informazioni, ciò è problematico: **più dati** vorremo registrare, **più dovremo pagare**. Occorre trovare una soluzione per registrare **pochi dati** ma utilizzabili per **numerosi controlli** in **breve tempo**. La soluzione è l'utilizzo di **accumulatori crittografici**.



Accumulatori crittografici

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

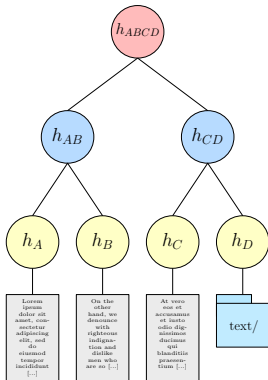
Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri



Strumenti che **comprimono molte informazioni** in una **costante** di dimensione fissa.

Un esempio ne sono i **Merkle Tree**, alberi binari in cui ogni foglia corrisponde all'hash di un elemento. Risalendo ogni nodo interno calcolerà il proprio hash con gli hash dei nodi figli, l'hash della root sarà **univoco** a quelle foglie in quell'ordine.



L'Obiettivo

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

Realizzare uno **strumento digitale distribuito** in grado,
tramite interazioni con **Git** e la **blockchain**, di:



Salvare hash di
repository su
blockchain



Esportare
sottoinsiemi di
repository verificabili



Verificare l'integrità
di singoli file e
repository



Il Software PineSU

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

PineSU è un software **Javascript** che sfrutta il run-time **Node.js**.

L'applicazione crea delle **strutture** sulle repository Git chiamate **Storage Unit (SU)** tramite metadati.

Queste SU sono le unità su cui effettueremo le singole operazioni, eccetto la registrazione su blockchain che si svolgerà collettivamente con l'ausilio di accumulatori crittografici.





Storage Unit

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

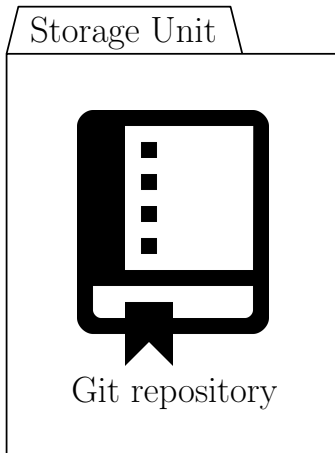
Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri



Metadati:

- Nome;
- Link repo remota;
- Descrizione;
- Nome;
- Visibilità;
- Data;
- Lista hash;
- Chiusura;



Workflow

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

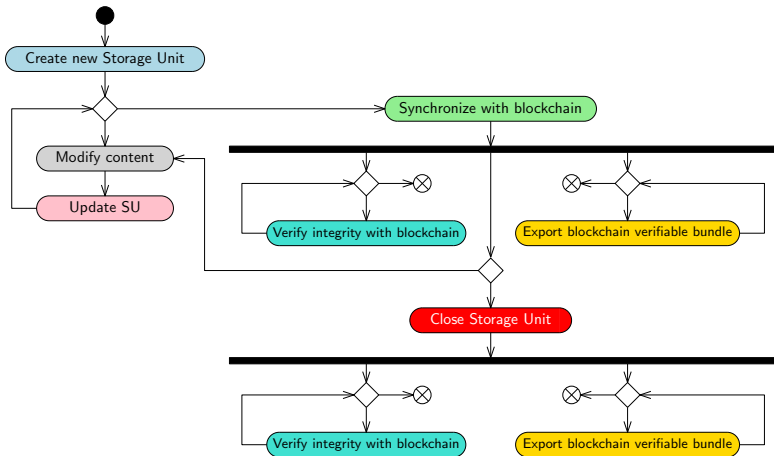
Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri





Ciclo vitale di una Storage Unit

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

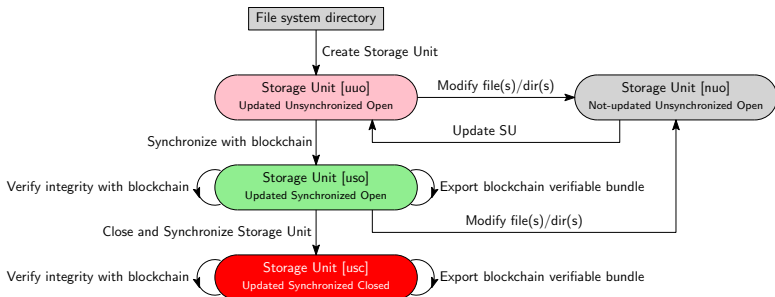
Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri





Architettura

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

II Problema

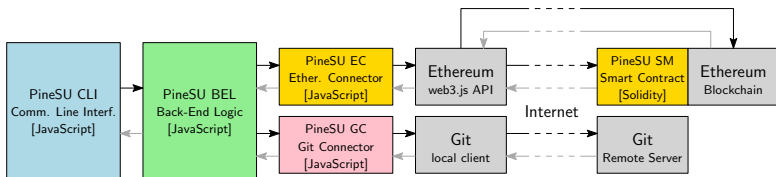
Concetti
preliminari

L'Obiettivo

II Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri





Architettura (Cont.)

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

- *PineSU CLI (Command Line Interface)*: **Crea l'interfaccia utente** con cui è possibile interagire e **richiama le funzioni** degli altri moduli all'occorrenza.
- *PineSU BEL (Back End Logic)*: Il **nucleo** di PineSU. **Gestisce le SU** e controlla la comunicazione con la **blockchain** e il client **Git** locale.
- *PineSU EC (Ethereum Connector)*: Si interfaccia con le **API della blockchain**.
- *PineSU GC (Git Connector)*: Si interfaccia con il **client Git**.
- *PineSU SM (Smart Contract)*: Permette **registrazioni permanenti** di singole SU nella blockchain.



PineSU CLI

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

```
MINGW64:/d/Progetti/Tesi

PineSU

? Welcome to PineSU, choose the operation to perform (Use arrow keys)
> Create new SU / Recalculate open SU
  Stage Storage Unit for Synchronization
  Close current SU
  Register Staged SUs in the blockchain network
  Check SU integrity
  Export files from current SU
  Check files integrity
(Move up and down to reveal more choices)
```



Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

Il nucleo centrale che si occupa di:

- 1 **Gestione dei file descrittori.**
- 2 **Gestione degli accumulatori crittografici.**
- 3 **Comunicazione con Git e blockchain.**



Gli accumulatori di PineSU

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

- *SU Merkle Tree*: Le sue **foglie** sono gli **hash dei file e directory** della SU. La sua **root** è l'**hash della SU** stessa.
- *Storage Group (SG)*: Le sue **foglie** sono le **SU da registrare** su blockchain nella prossima transazione.

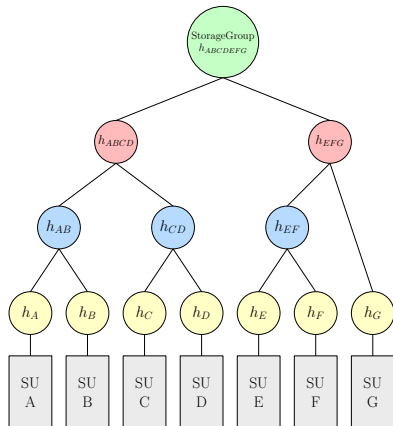


Figura: Uno Storage Group



Gli accumulatori di PineSU (Cont.)

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

- *Merkle Calendar (MC)*: Albero in cui le **foglie** sono i Blockchain Synchronization Point (**BSP**), istanze di Storage Group, a loro volta raggruppate in nodi rappresentanti **mesi e anni**, ciò rende i reperimenti di registrazioni passate più agevoli e veloci.

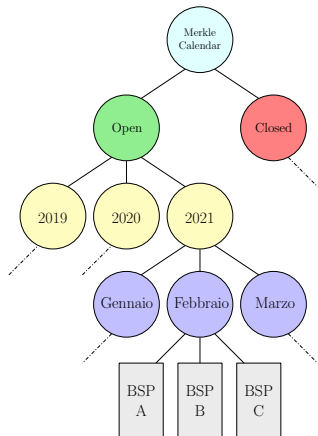


Figura: Un Merkle Calendar



Merkle Calendar UML

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

II Problema

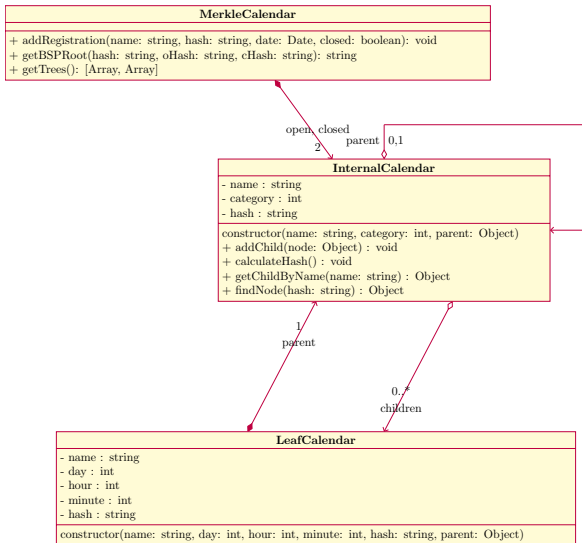
Concetti
preliminari

L'Obiettivo

II Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri





Codice - Reperimento di una BSP Root

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

```
for(let i = 0; i <= leafIndex; i++){
    leavesHash.push(monthNode.getChildByNum(i).getHash())
}
let newMonth = this.calculateHash(leavesHash);
let monthsHash = new Array();
for(let i = 0; i < monthIndex; i++){
    monthsHash.push(yearNode.getChildByNum(i).getHash())
}
monthsHash.push(newMonth);
let newYear = this.calculateHash(monthsHash);
let yearsHash = new Array();
for(let i = 0; i < yearIndex; i++){
    yearsHash.push(yearNode.getChildByNum(i).getHash())
}
yearsHash.push(newYear);
let newRoot = this.calculateHash(yearsHash)
```



Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

GitConnector

- git : SimpleGit

```
constructor(dir: string)
+ init() : void
+ add(arg: string) : void
+ commit(msg: string, enmsg: boolean) : void
+ getRepoFiles() : Array
+ push() : void
+ pull() : void
+ reset() : void
+ hasRemote() : Array
+ custom(commands: Array) : string
```



Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

EthConnector

- web3 : Web3
- w1 : string
- w2 : string
- k : string

```
constructor(host: string, w1 : string, w2 : string, k : string)
+ deploy(hashRoot: string) : string
+ verifyHash(transHash: string, hash: string) : boolean
```



Codice - Salvataggio di un hash su blockchain

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

```
async deploy(hashRoot){  
  const ct = await this.web3.eth.accounts  
    .signTransaction({  
      from: this.w1,  
      to: this.w2,  
      data: hashRoot,  
      gas: 3000000,  
    },  
    this.k  
  );  
  const receipt = await this.web3.eth  
    .sendSignedTransaction(ct.rawTransaction);  
  return receipt.transactionHash;  
}
```



PineSU SM

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

```
contract SURegistry {  
  
    string StorageUnit;  
    mapping(uint => string) public registry;  
    uint public SUCount;  
  
    function addSU(string memory hashSU) public {  
        SUCount++;  
        registry[SUCount] = hashSU;  
    }  
}
```

Codice dello **Smart Contract** che gestisce il salvataggio su blockchain delle **single SU**.



Le operazioni disponibili

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

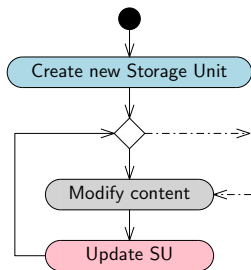
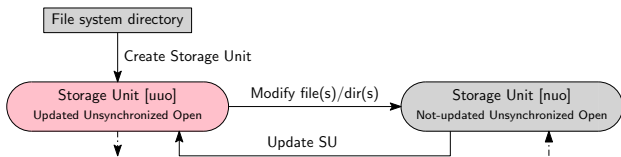
- 1 **Creazione** di una Storage Unit o **Ricalcolo** di una Storage Unit pre-esistente.
- 2 **Staging** di una Storage Unit nello Storage Group.
- 3 **Registrazione** dello Storage Group nella Blockchain.
- 4 **Chiusura** di una Storage Unit.
- 5 **Esportazione** di sottoinsiemi di file da una Storage Unit.
- 6 **Controllo** di integrità di **singoli file** esportati da altre Storage Unit.
- 7 **Controllo** di integrità su una **Storage Unit**.



Creazione di una Storage Unit

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli



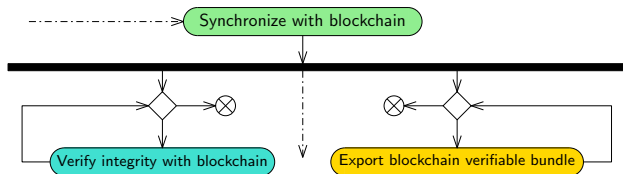
```
var merkleroot =  
  gitLogic.calculateTree(filelist);  
await inquirer.askSUDetails(  
  files.getCurrentDirectoryBase(),  
  remote).then((details) => {  
    details.owner = w1  
    details.hash = merkleroot  
    details.filelist = filelist  
    details.closed = false  
    files.saveJSON(details);  
  });
```



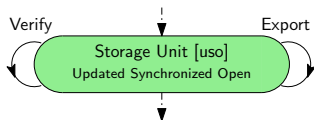

Registrazione di uno Storage Group

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli



```
var [doc, openRoot, closedRoot] =
  files.createSGTrees(sg);
ethLogic.
  addToTree(openRoot, mc, false);
ethLogic.
  addToTree(closedRoot, mc, true);
var [oHash, cHash, transHash] =
  await ethLogic.registerMC(mc);
for(var el of document){
  el.transHash = transHash;
  files.createReg(el);
}
```





Visualizzazione post-registrazione

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

BLOCK	MINED ON	GAS USED
94	2021-08-18 16:00:20	21320
BLOCK	MINED ON	GAS USED
93	2021-08-18 15:27:51	21320
BLOCK	MINED ON	GAS USED
92	2021-08-18 15:26:23	21320
BLOCK	MINED ON	GAS USED
91	2021-08-18 15:25:55	21320

VALUE	GAS USED
0.00 ETH	21320
TX DATA	
0xe67006f15ecd3fa2719d148be68d3a3242e1be8b	

EVENTS

← BACK		BLOCK 94	
GAS USED	GAS LIMIT	MINED ON	BLOCK HASH
21320	6721972021-08-18 16:00	0x6721972021-08-18 16:00	0xbaa08b6b640accbb9648ff313929998ced4a1abe10
5	0:20		4d6769756c
TX HASH			
0xc563030328e652d427cd00707d7a0e2ce0bcf6c76b23482469eb497e2dc87d2e			
FROM ADDRESS	TO CONTRACT ADDRESS	GAS USED	VALUE
0xCF2354abFC002905532D086bF647754A84	0x3a6990caE86a35a4022105b4c090DEF6490	21320	0
732966	8A0629		

```

"path":
  "D:/sample",
"root":
  "e67006f15ecd3[...]e6
  8d3a3242e1be8b",
"transHash":
  "0xc563030328e[...]69
  eb497e2dc87d2e"

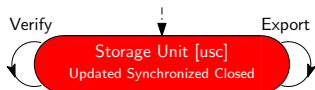
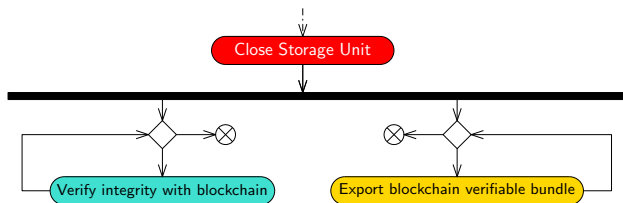
```



Chiusura di una Storage Unit

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli



```
if(fs.existsSync(".pinesu.json")){  
  var data =  
    fs.readFileSync(".pinesu.json")  
  var myObj = JSON.parse(data);  
  myObj.closed = true;  
  fs.writeFileSync(".pinesu.json",  
    JSON.stringify(myObj));  
  return myObj;  
}
```



Esportazione di sottoinsiemi di Storage Unit

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

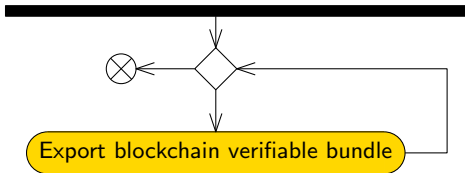
Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri



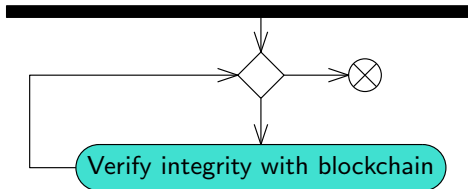
```
var zip = new AdmZip()
var fl = JSON.stringify(json)
zip.addFile(".pifiles.json", Buffer.alloc(fl.length, fl))
for(var el of list){
    zip.addLocalFile(path)
}
zip.writeZip("../pinesuExport.zip")
```



Controllo d'integrità su una Storage Unit

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli



Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

```
async verifyHash(transHash, hash){  
  const res =  
    await this.web3.eth.getTransaction(transHash)  
  if(res.input == "0x"+hash){  
    return true;  
  } else {  
    return false;  
  }  
}
```



Node.js

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

Node.js è un **ambiente di run-time**, che permette di eseguire codice **Javascript**.

Esso ha come obiettivi chiave l'**efficienza** e la **scalabilità**, può infatti eseguire velocemente codice Javascript sia **server-side** che **client-side**.

Parte fondamentale di Node sono i suoi numerosi **moduli**: librerie e framework realizzati dalla comunità e installabili con facilità tramite il package manager **npm**





Moduli dei connettori

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

web3.js è un **modulo npm** che permette di interagire con **nodi Ethereum** locali e remoti.

PineSU EC lo utilizza per effettuare le **transazioni** con i suoi wallet e per comunicare con lo **Smart Contract**.



Simple Git è un **modulo npm** che permette di comunicare con il **client Git** locale.

Usato in *PineSU GC*, esso permette l'esecuzione di comandi in maniera **asincrona**.



Altri moduli

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speciali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri



Inquirer.js è un **modulo npm** che facilita la creazione di **interfacce utente** tramite menù testuali.

In *PineSU CLI* viene usato per **interagire** con l'utente ponendogli **domande** dalla risposta chiusa o aperta.

ADM-ZIP è un **modulo npm** che consente di creare **cartelle compresse** in formato ZIP.

PineSU BEL lo utilizza per **esportare sottoinsiemi** di SU mantenendo la **struttura gerarchica** originale.





Sviluppi futuri

Condividere
informazioni
in modo
sicuro
combinando
Git e
Blockchain

Laureando:
Paolo Speziali
Relatore:
Luca Grilli

Il Problema

Concetti
preliminari

L'Obiettivo

Il Software
PineSU

Tecnologie
utilizzate

Sviluppi futuri

- 1 Migliorare gestione degli **accumulatori crittografici** per le **single SU**.
- 2 Implementare **Smart Contract** dalle migliori funzionalità.
- 3 Aggiungere **connettori** per **ulteriori blockchain**.
- 4 Creare **portale web** con **server Git** per la gestione remota delle SU (*ambizioso*).