



**UNIVERSITÀ  
DEGLI STUDI DI BARI  
ALDO MORO**

**DIPARTIMENTO DI INFORMATICA  
CORSO DI LAUREA MAGISTRALE IN SICUREZZA INFORMATICA**

---

**TESI DI LAUREA IN INFORMATICA FORENSE**

**ANALISI, PROGETTAZIONE,  
E REALIZZAZIONE  
DI UNA MACCHINA SPECIALIZZATA  
IN ACQUISIZIONE FORENSE**

*Relatore:*

Ch.mo Prof. Ugo LOPEZ

Laureando:

Angelo Oliva

---

ANNO ACCADEMICO 2020/2021



## ● Sommario

Sommario	3
1. Introduzione	5
1.1. La scienza forense	5
1.2. La branca dell'informatica forense	6
1.3. Rami dell'Informatica Forense	9
2. Acquisizione Forensi Memorie di Massa	11
2.1. Hardware per l'acquisizione forense	12
2.1.1. Tableau TX1	12
2.1.2. Falcon NEO	13
2.1.3. Atola Insight Forensic	15
2.1.4. Tableau TD2u	16
2.1.5. PC Standard	17
2.2. Software per l'acquisizione forense	18
2.2.1. Caine	18
2.2.2. Tsurgi	19
2.2.3. Tiny Core Forensic Edition	20
3. Analisi e Progettazione	22
3.1. Introduzione	22
3.2. Definizione del problema	23
3.3. Requisiti	23
3.4. Single Board Computer	24
3.5. Architettura ARM	27
3.6. Hardware SBC	28
3.6.1. Raspberry Pi 4	28
3.6.2. Odroid-N2	29
3.6.3. RockPro64	30
3.6.4. Banana Pi M4	31
3.6.5. Asus Tinker Board S	32
3.6.6. Aaeon PICO-WHU4	33
3.7. Software SBC	34
3.7.1. Flint OS	34
3.7.2. Raspberry Pi OS	34
3.7.3. DietPi	35
3.7.4. piCore	36
3.8. Hardware & Software Target	36
3.9. Casi d'uso	37
3.9.1. Acquisizione memorie USB	37
3.9.2. Acquisizione memorie SD	38
3.9.3. Acquisizione hard disk	38
4. Realizzazione	40
4.1. Assemblaggio	40
4.2. Installazione piCore	40
4.3. Write Blocker – Blockdev	43
4.4. Data Dump	46
4.5. Integrazione GUI	47
4.6. Supporto NTFS	48
4.7. Compilatore Python	49

4.8.	File Manager	49
4.9.	Browser	50
4.10.	Text Editor	51
4.11.	Ulteriori App	52
4.12.	RHash	55
4.13.	Wallpaper	56
4.14.	Script PiCore FE Install	58
4.15.	Remaster IMG	58
5.	Test Acquisizione	59
5.1.	Configurazioni	59
5.1.1.	Configurazione C1 – Caine/PC	59
5.1.2.	Configurazione C2 – PicoreFE/RaspberryPi4	60
5.2.	Device Sorgenti	60
5.3.	Device Destinazione	61
5.4.	Acquisizione C1 - Caine/PC	62
5.4.1.	Avvio Caine	62
5.4.2.	Abilitazione scrittura e montaggio	63
5.4.3.	Acquisizione	63
5.4.4.	Hash dell'immagine	64
5.5.	Acquisizione C2 - PicoreFE/RaspberryPi4	65
5.5.1.	Identificazione supporto di destinazione	65
5.5.2.	Abilitazione scrittura e montaggio	67
5.5.3.	Acquisizione	68
5.5.4.	Hash dell'immagine	68
5.5.5.	Riepilogo acquisizione	69
6.	Risultati	70
6.1.	piCore Forensic Edition	70
6.2.	Test acquisizione	70
7.	Sviluppi Futuri	72
8.	Bibliografia	74

# 1. Introduzione

## 1.1. *La scienza forense*

“La scienza forense è l'applicazione di tecniche e metodologie scientifiche alle tradizionali investigazioni di carattere giudiziario, in relazione all'accertamento di un reato o a un comportamento sociale. Il suo scopo è quello di identificare, preservare, recuperare, analizzare e presentare le prove nel corso di un'indagine.

Nell'uso tradizionale, il termine forense indica sia una forma di evidenza giuridica sia una categoria di pubblicizzazione legale. Nell'uso moderno, il termine forense rispetto a scienza forense può essere considerato come un sinonimo di legale o, comunque, sotteso al sistema penale.” (1)

Sebbene trovi applicazione sin da prima dell'Impero Romano il “padre” della moderna scienza forense è Alphonse Bertillon. Alla fine del diciottesimo secolo questo ufficiale di polizia francese, oltre a introdurre un sistema di documentazione fotografica della scena del crimine, inventò l'antropometria giudiziaria, nota anche come sistema Bertillon, che era basata sull'analisi delle misurazioni fisiche. (2)

Oltre all'invenzione della fotocamera, si scoprì come identificare la presenza di sangue mediante l'utilizzo del perossido di idrogeno.

Il regno della criminologia viene profondamente rinnovato grazie a tutti questi progressi rapidi nel mondo della scienza che preparano la strada verso successive innovazioni.

Gli inglesi Henry Faulds e William Herschel e lo scienziato americano Thomas Taylor determinarono un significativo passo avanti nella scienza forense, dettagliando l'unicità delle impronte digitali umane e la potenzialità nel loro uso per giungere alla codificazione e alla standardizzazione delle pratiche interne alla scienza forense. Enormi contributi a tale scienza diede

il dott. Edmond Locard, scienziato e criminologo francese, esperto in legge e medicina. Egli propose un principio, che ancora oggi è valido nelle indagini sulla scena del crimine, e cioè che "tutto lascia una traccia". Da tale principio, detto di scambio, deriva l'idea che tutti quelli che entrano in una scena del lasciano dietro di sé alcune tracce che possono contribuire alla formazione di prove. Allo stesso modo, tutti i presenti, quando vanno via, portano con sé tracce della scena del crimine.

All'inizio del diciannovesimo secolo, precisamente nel 1910, le convinzioni e le ricerche di Locard convincono il dipartimento di polizia di Lione, in Francia, a fornirgli un ufficio e uno staff per analizzare le prove raccolte dalle scene dei crimini. Le due stanze dell'attico e i suoi due assistenti divennero presto il primo laboratorio del crimine al mondo chiamato The First Crime Lab. (3)

Il perfezionamento delle tecniche, sia nell'analisi che nella conservazione delle prove fece grandi progressi nel XX secolo, basandosi naturalmente sui fondamenti del XIX secolo, alla fine del quale dal ruolo fondamentale delle impronte digitali si giunse alla analisi e identificazione del DNA.

L'uso recente del DNA nelle indagini penali ha reso possibile non solo l'identificazione di numerosi criminali, ma anche la revisione di sentenze di condanna precedenti e la conseguente liberazione di centinaia di persone innocenti. Con i nuovi progressi nella tecnologia della polizia e nell'informatica, le indagini sulla scena del crimine e le scienze forensi diventeranno solo più precise mentre più tardi si svilupperà una nuova importante branca, quella della scienza digitale forense.

## **1.2.    *La branca dell'informatica forense***

“L'informatica forense è una branca della scienza digitale forense legata alle prove acquisite da computer e altri dispositivi digitali. Il suo scopo è quello

di esaminare dispositivi digitali seguendo processi di analisi forense al fine di identificare, preservare, recuperare, analizzare e presentare fatti o opinioni riguardanti le informazioni raccolte.” (4)

I suoi obbiettivi sono:

- Cristallizzare le sorgenti degli elementi di prova;
- Riconoscere e identificare dati e informazioni utili;
- Estrarre e catalogare dati e informazioni utili;
- Comparare dati e informazioni utili tra di loro;
- Relazionare scientificamente sul lavoro svolto.

Essa si sviluppa a partire dai primi anni '80, quando per i consumatori diventano accessibili i primi personal computer, che in alcuni casi cominciano ad essere utilizzati per applicazioni criminali. Quindi nascono i primi crimini identificati e riconosciuti come crimini informatici che registrano una crescita esponenziale nel corso degli anni, basti pensare che i casi tra il 2002 e il 2003 hanno registrato un aumento del 67%.

Si rende necessaria una nuova metodologia per recuperare e analizzare le prove digitali per utilizzarle di fronte ad una corte. Questo ha portato l'informatica forense a non essere più utilizzata esclusivamente per l'analisi della prove ma a diventare anche uno strumento dei principali mezzi di investigazione per diversi crimini.

Un crimine informatico è un fenomeno criminale che si caratterizza nell'abuso della tecnologia informatica sia hardware che software, per la commissione di uno o più crimini e abbraccia un ampio ventaglio di attività suddivise in due categorie; la prima si caratterizza per l'uso della tecnologia informatica per compiere l'abuso come per esempio l'utilizzo Spam o Malware. Nella seconda categoria l'utilizzo dell'elaboratore è un mezzo per la commissione del reato come per esempio pedopornografia, cyberstalking, frodi informatiche, spionaggio, cyberbulling, Phishing.

Nei processi civili nei quali tale scienza è coinvolta, aumentano quindi le cosiddette "computer generated evidence" che consentono l'applicabilità dell'informatica forense anche nei casi di reati non strettamente legati all'informatica come per esempio omicidi e stupri, ma anche in contesti differenti come quello civile, amministrativo e tributario.

Salvaguardare i dati presenti sui dispositivi di archiviazione di massa posti sotto il sequestro, cioè non più nella disponibilità del proprietario, è un aspetto fondamentale dell'informatica forense.

Gli operatori, che analizzano i dispositivi di archiviazione, per garantire l'inalterabilità dei dati, utilizzano precise metodologie per certificare la corrispondenza esatta dei contenuti in qualsiasi fase dell'analisi. Per fare ciò devono "cristallizzare il dato", ossia mettere in atto procedimenti tecnologici adeguati ad impedire scritture, anche accidentali dei singoli bit e a consentire, anche in un momento successivo che i dati presenti non siano mutati. Per rispondere a tali criteri, oltre ad utilizzare strumenti hardware e software che impediscano qualsiasi scrittura sui dispositivi di archiviazione impiegano algoritmi di Hash in modo che ciascun file o tutto il contenuto del dispositivo generi una specie di impronta digitale, in modo da verificarne l'integrità in ogni fase successivo il sequestro.

Un'analisi forense utilizza strumentazione che garantisca la non alterazione del reperto e le relative garanzie in merito alla catena di custodia. Questo termine si riferisce alla documentazione cronologica o alla traccia cartacea che mostra il sequestro, la custodia, il controllo, il trasferimento, l'analisi, e la disposizione di elementi di prova, fisica o elettronica. (5)

I pilastri nella costruzione della prova scientifica sono conosciuti universalmente come i criteri Daubert (6):



- **Attendibilità:** La stessa ricerca fatta da un altro ricercatore con gli stessi metodi, con gli stessi soggetti avrebbe ottenuto i medesimi risultati?
- **Validità:** I risultati rispecchiano lo stato delle cose?
- **Generalizzabilità:** I risultati sono applicabili a casi analoghi?
- **Credibilità:** La procedura e i risultati della ricerca sono affidabili?
- **Falsificabilità:** Sono stati testati scientificamente?
- **Blind Per Review:** Sono stati sottoposti ad una revisione critica?
- **Accettabilità:** Sono condivisi dalla comunità scientifica?
- **Controllo Metodologico:** Si conosce il loro potenziale di errore?
- **Affidabilità:** È stata deliberata la stabilità della misura in tempi diversi?
- **Validità:** Si conosce l'accuratezza dello strumento, ovvero il grado con cui esso misura quello per il quale è stato costruito per misurare?
- **Validità Incrementale:** Si conosce di quanto la validità della valutazione sia aumentata alla luce dell'aggiunta di nuove informazioni a quelle precedentemente o tradizionalmente considerate?
- **Sensitività:** Si conosce l'incidenza dei falsi positivi?
- **Specificità:** Si conosce l'incidenza dei falsi negativi?

### 1.3. ***Rami dell'Informatica Forense***

A seconda degli ambiti su cui opera e degli strumenti utilizzati possiamo distinguere una serie di rami dell'informatica forense che principalmente sono:

- **Computer forensics:** ramo dell'informatica forense che si occupa dell'analisi forense di computer, server e simili;

- **Mobile forensics:** ramo dell'informatica forense che si occupa dell'analisi forense di smartphone e simili;
- **Network forensics:** ramo dell'informatica forense che si occupa dell'analisi forense di reti di calcolatori;
- **Incident response forensics:** ramo dell'informatica forense che si occupa dell'analisi forense di un sistema compromesso;
- **Open Source Intelligence (OSInt):** ramo dell'informatica forense che si occupa della raccolta di informazioni attraverso l'utilizzo di fonti di pubblico accesso;
- **Cloud forensics:** ramo dell'informatica forense che si occupa dell'analisi forense di un'infrastruttura cloud;
- **Bitcoin forensics:** ramo dell'informatica forense che si occupa dell'analisi forense di un sistema di criptovalute;
- **Drone forensics:** ramo dell'informatica forense che si occupa dell'analisi forense di indagini digitali su diversi elementi del volo e della programmazione dei droni.

## 2. Acquisizione Forensi Memorie di Massa

Lo sviluppo tecnologico ha coinvolto i computer, ma anche tutti gli apparati informatici nell'illecito. Infatti in molti casi, sempre in aumento, questi strumenti tecnologici vengono utilizzati per commettere reati. Le prove, spesso non relative esclusivamente a reati informatici, si trovano su vari dispositivi di memorizzazione come per esempio CD, DVD, Hard Disk, Pen Drive ecc...

Le informazioni in informatica, pur essendo immateriali, sono racchiuse nei vari file che devono essere memorizzati necessariamente in una memoria di massa, come quelle sopra elencate. Scegliere l'oggetto del provvedimento al momento del sequestro è una scelta ardua e complessa. Normalmente vengono confiscate solo ed esclusivamente le memorie di massa, ma non è da escludere che in casi particolare può essere ordinato di sequestrare l'intero apparato informatico.

Per questo motivo vogliamo dedicare la nostra attenzione, in modo particolare, sugli strumenti per acquisire le memorie di massa comuni. Il nostro target saranno quindi gli hard disk, le pen drive usb e le schede sd.

Il processo di acquisizione della prova si suddivide nelle seguenti fasi (7):

- **Identificazione:** Questa prima fase ha come scopo quello di identificare i supporti da cui estrarre i dati da analizzare, nonché eventuale trasporto sicuro nel laboratorio forense;
- **Acquisizione:** Consiste nella duplicazione forense del supporto da analizzare con congelamento del dato acquisito tramite funzioni di hash;
- **Analisi:** In questa fase, i contenuti delle copie forensi vengono analizzati attraverso appositi strumenti forensi;

- **Presentazione:** La parte terminale del processo consente all'informatico forense di produrre un report dettagliato sulle ricerche effettuate e sui risultati raggiunti.

Il cammino che stiamo per intraprendere riguarderà la fase di acquisizione; analizzeremo quindi strumentazione hardware e software fondamentale per questa fase.

## **2.1. *Hardware per l'acquisizione forense***

In ogni fase di una analisi forense approfondita ed esauriente è necessaria una specifica strumentazione hardware e software che varia in base a vari parametri.

Nel nostro lavoro ci occuperemo solo di alcuni strumenti specifici per l'acquisizione da memorie di massa.

Dal punto di vista hardware distinguiamo due categorie. Quella di prodotti ad hoc, progettati e realizzati per l'acquisizione forense, molto costosi e performanti. Questi si rendono necessari in laboratori avanzati e specializzati e sono in grado di eseguire acquisizioni simultanee. In essi è implementato un write blocker hardware per evitare l'alterazione accidentale di qualsiasi memoria collegata. Generalmente questo sistema è progettato a prova di errore; infatti tutti i dispositivi da acquisire sono collegati nella parte sinistra dello strumento, che è quella che ha le porte protette in scrittura.

L'altra ipotesi è quella di utilizzare un hardware equipaggiato con sistemi operativi forensi e/o con software forensi specializzati. Questi con costi ridotti consentono di acquisire memorie in modo ufficiale e certificato.

### **2.1.1. Tableau TX1**

Il TX1 è l'ultimo duplicatore di marca Tableau e permette di raggiungere le prestazioni più elevate sul mercato. È in grado di acquisire due diversi

dispositivi senza perdita di performance su sei diverse destinazioni. Consente connessioni con SATA, SAS, PCIe, USB 3.0, FIREWIRE. Oltre a due processi attivi, consente di accodare altre attività che verranno avviate automaticamente. Oltre a queste funzionalità, dispone di una scheda di rete a 10Gb che lo rende adatto anche per acquisizioni via rete. (8)



Prezzo medio: 3000€.

### **SPECIFICHE:**

- Immagini SATA, USB 3, PCIe, SAS, FireWire 800, IDE ed acquisizioni di rete;
- Outputs verso SATA, USB 3, SAS e scheda di rete 10Gb;
- Supporta fino a due operazioni contemporanee attive (acquisizione, verifica hash, wiping);
- Supporta fino a quattro destinazioni contemporanee;
- Possibilità di inserire job in coda "acquisizione, verifica hash, wiping";
- Confezione in scatola, disponibile come opzione valigia Pelican;
- Il dispositivo dispone di uno screen touch a colori.
- Aggiornamenti gratuiti del firmware Tableau.

### **2.1.2. Falcon NEO**

Forensic Falcon-NEO, è uno strumento fortemente specializzato per soluzioni di imaging forense, e può raggiungere velocità di imaging di decine di GB per minuto. Il Falcon-NEO può acquisire immagini da un massimo di 5 unità sorgente fino a un massimo di 9 destinazioni contemporaneamente

per fornire una raccolta di prove digitali efficiente e sicura. La soluzione può creare immagini da/verso un repository di rete con due porte 10GbE. Il Falcon-NEO orientato al futuro è progettato per semplificare il processo di raccolta delle prove.

Il Falcon NEO raggiunge velocità di imaging superiori a 50 GB/min. La raccolta di prove digitali efficiente e sicura viene realizzata con un set di funzionalità che fornisce funzionalità sofisticate con l'obiettivo di ridurre i tempi di acquisizione. Progettato per soddisfare i futuri progressi tecnologici nel campo della medicina legale, il Falcon-NEO stabilisce nuovi standard nella tecnologia di imaging forense.

Per facilitare il lavoro degli investigatori, Falcon Neo può acquisire immagini direttamente da portatili PC e Mac utilizzando iSCSI boot client oppure in Target Disk Mode, risparmiando il tempo di rimuovere gli hard disk. Le porte della sorgente bloccata in scrittura includono: 2 SAS/SATA, 1 USB 3.0 and 1 4-lane PCIe. Le porte di destinazione disponibili sono: 2 SAS/SATA, 2 SATA, 1 USB 3.0 and 1 4-lane PCIe. (9) SCSI, FireWire mentre la Fibre Channel è supportata da moduli opzionali, come l'opzione della card Thunderbolt I/O che permette di catturare dati da e per Thunderbolt 3/USB-C. (10)



Prezzo medio: 5000€.

## **SPECIFICHE:**

- Imaging oltre i 50 GB / min;
- Clonazione da PCIe a PCIe con una velocità di 90 GB / min;
- Imaging + verifica da 5 sorgenti a 9 destinazioni 2 connessioni di rete 10GbE;
- Cattura precisa dei file, per ridurre i tempi di acquisizione e migliorare la qualità;
- Possibilità di verificare l'immagine già durante l'acquisizione;
- Funzionalità di acquisizione del traffico di rete. Cattura del traffico di rete, VOIP, attività Internet;
- I / O card integrate per supportare le nuove tecnologie non appena arrivano sul mercato  
Imaging da più fonti contemporaneamente:  
l'acquisizione di immagini da 5 sorgenti a 9 destinazioni può essere svolta senza che la qualità ne risenta.

### **2.1.3. Atola Insight Forensic**

Atola Insight è un sistema di recupero dati e forense all-in-one per uso professionale. È stato progettato per l'uso sia in laboratorio che sul campo e offre complesse funzioni di recupero dei dati, che sono state brillantemente automatizzate, insieme a utilità per l'accesso manuale ai dischi rigidi al livello più basso. Il suo sofisticato software è racchiuso in un'interfaccia utente semplice ed efficace. I punti di forza di questo prodotto sono le sue diverse funzionalità, l'alta usabilità di utilizzo e i forti canali di supporto del prodotto.

Atola Insight è sviluppato da un team di rinomati ingegneri del settore del recupero dati in collaborazione con le forze dell'ordine e gli esperti forensi di tutto il mondo.

Un'interessante funzionalità per le acquisizioni forensi è quella che permette di eseguire tre sessioni di imaging simultanee su un'ampia gamma di supporti. (11)



Prezzo medio: 2000€.

### **SPECIFICHE:**

- 3 simultanee sessioni di imaging + multi-tasking
- 2 porte integrate Ethernet 10Gb;
- velocità nelle sessioni di imaging fino a 500 MB/s
- E01, AFF4 o Raw target immagini create dalla rete o da drive direttamente collegati;
- Creazione della image contemporaneamente su tre drive;
- Supporta SATA, IDE, USB;
- Estensioni: SAS, Apple PCIe (2013 - recent models), NVMe and M.2 PCIe SSDs;
- Integra un hardware write blocker per tutte le porte sorgenti.

#### **2.1.4. Tableau TD2u**

Tableau Forensic Duplicator è stato creato per eccellere sia in ambienti di campo che di laboratorio. Si tratta di un duplicatore forense completo che offre la combinazione ideale di facilità d'uso, affidabilità e alte prestazioni, in termini di velocità, nelle creazioni di immagini forensi di dischi rigidi e unità a stato solido.

Il prodotto si è anche guadagnato il premio di miglior Hardware forense nel 2016 "2016 Computer Forensic Hardware of the Year". (12)

L'apparecchio supporta tutte le interfacce per acquisire memorie di massa, quali SATA, USB 3.0 e IDE.

Il TD2U permette di creare fino a TRE COPIE del supporto da acquisire (due dischi SATA e un disco in USB 3.0)





Prezzo medio: 2000€.

### **SPECIFICHE:**

- Disk-to-Disk (clone) duplicazione;
- Disk-to-File (image) duplicazione;
- Formattazione;
- Wipe;
- Hash (MD5 or SHA-1);
- HPA/DCO rilevamento e rimozione;
- Blank Disk Check;
- Massima velocità 15GB/M.

### **2.1.5. PC Standard**

Gli strumenti hardware forensi realizzati ad-hoc costituiscono sicuramente l'ipotesi migliore per eseguire delle acquisizioni in modo rapido ed efficiente. Dato il prezzo elevato sono sicuramente convenienti in laboratori specializzati in analisi forense che acquisiscono grandi quantità di supporti. Tuttavia, una soluzione, sicuramente valida e anche meno costosa consiste di utilizzare un classico personal computer. Per far ciò, oltre a utilizzare un hardware performante, occorre scegliere e caricare un sistema operativo specializzato nell'ambito forense. Questo non deve essere necessariamente installato, ma può essere avviato tramite in modalità live impostando il boot

da memoria usb. Analizzeremo di seguito i sistemi operativi specializzati in acquisizione e analisi forense.

## **2.2.     *Software per l'acquisizione forense***

Nella letteratura troviamo vari tool open source presenti. Un esempio sono le distribuzioni forensi Caine o Tsurgi. La loro principale caratteristica è quella di aver configurato un write blocker a basso livello software per evitare qualsiasi alterazione di un singolo bit sul dispositivo da acquisire. Analizziamo ora nel dettaglio le caratteristiche di queste distro Linux forensi.

### **2.2.1. Caine**

Acronimo di Computer Aided INInvestigative Environment è una distribuzione live Linux italiana gestita da Giovanni Bassetti. Il progetto è iniziato nel 2008 come ambiente per promuovere l'analisi forense digitale, con diversi strumenti correlati preinstallati.

Questa piattaforma forense professionale integra strumenti software come moduli insieme a potenti script in un ambiente di interfaccia grafica. Il suo ambiente operativo è stato progettato con l'intento di fornire al professionista forense tutta la strumentazione necessari per eseguire il processo di indagine nelle varie fasi quali conservazione, raccolta, esame e analisi. Questo progetto è completamente aperto in piena linea con i principi Open Source. CAINE è una distribuzione Linux live, quindi può essere avviata da un supporto rimovibile (unità flash) o da un disco ottico ed eseguita in memoria. Può anche essere installato su un sistema fisico o virtuale. In modalità Live, il sistema può operare su oggetti di archiviazione dati senza dover avviare un sistema operativo di supporto. L'ultima versione 11.0 può essere avviata su UEFI/UEFI+Secure e Legacy BIOS consentendo l'utilizzo di CAINE su

sistemi informativi che avviano sistemi operativi precedenti (ad es. Windows NT) e piattaforme più recenti (Linux, Windows 10).

Il progetto è basato su Ubuntu 18.04 a 64 bit, utilizzando il kernel Linux 5.0.0-32. I requisiti di sistema per l'esecuzione come disco live sono simili a Ubuntu 18.04. Può essere eseguito su un sistema fisico o in un ambiente di macchina virtuale come VMware Workstation.

La distribuzione CAINE Linux ha numerose applicazioni software, script e librerie che possono essere utilizzate in un ambiente grafico o a riga di comando per eseguire attività forensi. Può essere utilizzato per eseguire analisi dei dati di oggetti dati creati su Microsoft Windows, Linux e alcuni sistemi Unix. Una delle principali funzionalità forensi dalla versione 9.0 è che imposta tutti i dispositivi a blocchi per impostazione predefinita in modalità di sola lettura. Il blocco della scrittura è una metodologia fondamentale per garantire che i dischi non siano soggetti a operazioni di scrittura da parte del sistema operativo o di strumenti forensi. Ciò garantisce che gli oggetti dati allegati non vengano modificati, il che avrebbe un impatto negativo sulla conservazione forense digitale.

Analisi forense di rete, database, memoria è resa possibile grazie a vari strumenti software forniti da CAINE. L'analisi dell'immagine del file system di NTFS, FAT/ExFAT, Ext2, Ext3, HFS e ISO 9660 è possibile tramite riga di comando e tramite il desktop grafico. L'esame di Linux, Microsoft Windows e alcune piattaforme Unix è integrato. Tramite il tool Guymager è possibile importare immagini del disco in formato raw (dd) e Expert Advanced (Exx). (13)

### **2.2.2. Tsurugi**

Questo sistema operativo è sviluppato da un gruppo completamente italiano. Rilasciato nel 2018 dal Tsurugi Linux team, guidato da Giovanni Rattaro,

questo ambiente consente di effettuare attività di digital forensics, analisi di malware e OSINT (Open Source Intelligence) (14).

Il sistema con codice open source e di distribuzione gratuita si presenta in tre distinte versioni, a seconda dell'utilizzo che sono:

- **Tsurugi Lab:** distribuzione da installare sulla postazione di lavoro dell'informatico forense;
- **Tsurugi Acquire:** sistema operativo dedicato all'acquisizione forense di dati che va avviato sul computer "bersaglio" e serve per acquisirne i contenuti per poi poterli analizzare con tutta calma sulla propria postazione di lavoro;
- **Bento:** utilities per effettuare indagini direttamente sul pc bersaglio (Windows, MAC e Linux).

Gli strumenti di computer vision sono utili ad esaminare immagini e video mediante un confronto diretto che altri file grafici della sospettata vittima oggetto del reato. Nei reati di Bitcoin forensics o cyberstalking questa funzionalità particolare integrata nel sistema operativo risulta molto utile.

È consentita anche la gestione remota della piattaforma consentendo attività di accesso a distanza.

Tutti i device sono connessi di default in modalità di sola lettura attraverso un write blocker posto a livello del kernel. Tsurugi Linux assicura la funzionalità di write blocking anche nella versione installata. (15)

### 2.2.3. Tiny Core Forensic Edition

TinyCore FE è un recente progetto molto interessante nel campo dell'informatica forense avviato nel 2020 da Ruffo Sara. Per questo sistema operativo minimale è stato implementato un Write Blocker chiamato BlockDev. Nell'ambito di analisi forensi è fondamentale garantire l'immodificabilità delle informazioni e questo strumento permette la

protezione dei dati presenti su supporti di memorizzazione oggetto dell'attività forense. I Write Blocker prevengono il rischio di modifiche sul supporto digitale oggetto dell'investigazione per evitare di invalidare potenziali prove involontariamente, assicurandone la connessione in read-only.

Il BlockDev utilizzato è di tipo Software based ovvero che agisce a basso livello quindi al di sotto del livello del filesystem e dei degli altri driver del dispositivo, assicurando quindi l'immodificabilità dei dati presenti sui supporti che si andranno ad analizzare.

Questo blocco ovviamente non è irreversibile; infatti attraverso un comando specifico è possibile riconvertire la partizione nuovamente in scrittura. (16)  
Inoltre, include altre applicazioni utili in ambito forense come RHash e Datadump.

## 3. Analisi e Progettazione

### 3.1. *Introduzione*

Prima di scegliere la strumentazione necessaria per poter eseguire delle acquisizioni bisogna tenere in considerazione due fattori essenziali quali efficienza e costi.

Dotarsi semplicemente di un computer potente non è sufficiente, occorre valutare anticipatamente le molte esigenze al fine di scegliere i corretti hardware e software, che in alcuni casi devono essere molto specifici.

Un altro aspetto fondamentale nella scelta è che la soluzione individuata possa garantire una buona scalabilità e permetta nel futuro delle espansioni.

Nella fase di acquisizione la velocità è un aspetto fondamentale, sia per questioni pratiche nella realizzazione della copia forense ma anche per ridurre al minimo qualsiasi tipo di alterazione del dispositivo. Ormai le memorie stanno diventando sempre più grandi con un ritmo sfrenato, raggiungendo anche diversi Terabyte; quindi, è bene ottimizzare tutte le risorse per rendere più rapida possibile questa fase per le proprie esigenze al fine di migliorare le performance. Questo aspetto si rende ancor più necessario nel caso di accertamenti irripetibili dove è indispensabile fare almeno due copie della memoria per essere sempre resilienti ad eventuali malfunzionamenti o guasti. Infatti, in alcuni casi può anche capitare che i dispositivi sono posti a sequestro per un brevissimo tempo, scaduto il quale devono essere restituiti ai soggetti possessori.

Al contrario nella fase di analisi, la velocità è relativa. Il tecnico, per studiare e organizzare i risultati ottenuti, dovrà procedere con cautela. Certo hardware e software prestanti sono sempre preferibili ma non fondamentali.

Gli hardware analizzati realizzati ad hoc per l'acquisizione forense sono molto costosi e non adatti ad un'iniziale necessità di acquisire le memorie di massa.

### **3.2. *Definizione del problema***

Ci si propone di realizzare una macchina specializzata in acquisizione forense utilizzando un hardware low cost, che permetta di acquisire la maggior parte delle memorie di massa utilizzate dagli utenti standard. Questa dovrà essere equipaggiata con un sistema operativo che consenta le funzioni di acquisizione. Infine, verranno eseguiti dei test di acquisizione al fine di testarne le prestazioni. Queste verranno confrontate con una configurazione comune e collaudata.

### **3.3. *Requisiti***

Le configurazioni della macchina da realizzare dovranno soddisfare i seguenti requisiti:

- Hardware Low Cost;
- Hardware Prestante;
- Hardware compatto e portatile;
- Write Blocker software;
- Sistema Operativo minimale con funzioni base di acquisizione;
- Tool di Hash dei dati.

I PC seppur molto più economici rispetto agli strumenti di acquisizione forense realizzati ad hoc, comunque rappresentano una possibile configurazione molto utilizzata nello scenario delle acquisizioni forensi. Dato che uno dei prerequisiti che ci siamo preposti richiede l'utilizzo di un hardware economico, vogliamo sperimentare una nuova tipologia di

macchine, che stanno prendendo piede negli ultimi anni, ovvero i cosiddetti computer single board.

Questi, pur essendo molto compatti, riescono ad offrire prestazioni ottimali. Infatti, questa categoria di computer è molto utilizzata per progetti customizzati come particolari media server, applicazioni domotiche e in ambito IoT.

### **3.4.     *Single Board Computer***

L'acronimo SBC, dall'inglese computer a scheda singola, identifica un Single-board computer. Questa macchina è completamente costruita su una scheda madre a circuito singolo equipaggiata con tutti gli elementi fondamentali di un computer. Infatti essa ingloba tutti i vari componenti indispensabili. Troviamo in primis un microprocessore, una memoria RAM, porte di input e output e altre funzionalità richieste per un computer funzionale. .

Essi si contraddistinguono principalmente dai classici e diffusissimi personal computer desktop per la caratteristica di non basarsi sui classici slot di espansione per funzioni aggiuntive, ma su connettori specifici, come per esempio i GPIO, per interfacciarsi con hardware e circuiti esterni. Inoltre, nella maggior parte dei casi utilizzano microprocessori a 8 e 16 bit e RAM statica.

Negli ultimi anni i SBC hanno subito una vera rivoluzione dotati di una prestantza e capacità computazionale molto elevata; il loro cuore pulsante è una cpu con architettura definita ARM a 32 o 64 bit. Infatti modelli più recenti hanno capacità simili, se non superiori per specifici utilizzi, a quelle classici notebook o dei diffusissimi tablet. Inoltre troviamo anche modelli molto compatti, utilizzati come server blade, che presentano prestazioni simili ai classici computer server.



La creazione di questa particolare specie di computer a singola scheda è stata concretizzata e resa possibile grazie allo sfrenato e spropositato sviluppo tecnologico che ha permesso un incremento della densità e l'integrazione dei transistor in un singolo microprocessore.

Questo fenomeno era stato enunciato sin dal 1965 da Moore, imprenditore e informatico statunitense cofondatore della Fairchild Semiconductor nel 1957 e dell'Intel nel 1968.

Egli ipotizzò che il numero di transistori nei microprocessori sarebbe raddoppiato ogni anno. L'ipotesi si rivelò vera solo dal 1975 al 1980. Invece negli anni Ottanta il periodo di tempo si allungò a due anni fino alla formulazione finale che a partire dagli anni Novanta fino ai nostri giorni ha bloccato il periodo a 18 mesi. Questa legge è diventata il metro e l'obiettivo di tutte le aziende che operano nel settore come Intel e AMD, ma anche dalle nuove case produttrici di smartphone, tablet e dei computer single board.

(17)

I vantaggi complessivi di una configurazione a scheda singola sono quelli di attenuare i costi complessivi di un sistema, e di eliminare qualsiasi tipo di connettore tra componenti grazie alla riduzione del numero di schede a circuito stampato necessarie. Raggruppando in un'unica scheda più funzioni si ottiene complessivamente un sistema più piccolo ed efficiente similmente come avviene nei classici notebook. I connettori, slot di rame e schede di espansione sono una fonte frequente di problemi di affidabilità; quindi, un sistema a scheda singola elimina queste debolezze.

Le applicazioni degli SBC sono utilizzate per svariati generi come slot machine, video poker e controllo di automazione delle macchine vending (distributori automatici). Questo è facilmente personalizzabile grazie alla presenza, direttamente sulla scheda madre, di uno slot di I/O più mirato a un'applicazione industriale, come gli I/O digitali e analogici integrati. Questi

possono essere direttamente interfacciati con dei relè di comando per alimentare/disalimentare delle utenze a partire da luci fino alla movimentazione sincronizzata di robot.

I computer a scheda singola, negli ultimi tempi, sono più comunemente utilizzati anche nella domotica casalinga, dove possono controllare le utenze come luci, elettroserrature, climatizzazione o irrigazione. La loro applicazione si espande in situazioni industriali in cui vengono utilizzati in formato rack per il controllo di processo di produzione o interfaccia ai macchinari industriali. Infine, trovano spazio anche in applicazioni estreme come l'esplorazione di acque profonde, nello spazio, sui razzi e sullo Space Shuttle.

Rispetto ai computer classici gli SBC presentano un'efficienza energetica maggiore grazie alla riduzione dei vari connettori e componenti che li rendono nel complesso più leggeri e piccoli grazie al processo di integrazione.

Inizialmente il vantaggio principale di una scheda madre classica rispetto a un SBC era il costo. Questo perché non erano diffusi come le schede madri ATX prodotte in milioni di pezzi e destinate a mercati differenti bensì erano indirizzati ad una piccola nicchia di mercato con produzione ridotta che dettava quindi un prezzo maggiore. Essendo gli SBC completamente integrati, un guasto di un qualsiasi componente porterà alla sostituzione di tutto il minicomputer. (18)

Negli ultimi anni lo scenario si è notevolmente modificato infatti sul mercato sono stati prodotti modelli commerciali che hanno un costo nettamente inferiore ai classici personal computer con potenza computazionale molto elevata.

### 3.5. *Architettura ARM*

Un altro interessante aspetto da approfondire riguarda l'architettura della CPU di questi computer a scheda singola. Infatti, utilizzano una nuova architettura denominata ARM invece delle classiche x86 o x64 che siamo abituati ad utilizzare.

In realtà il progetto ARM iniziò nel 1983 nella sezione ricerca e sviluppo della Acorn Computers Ltd. Il primo prototipo chiamato ARM1 venne ultimato dal team nel 1985 e l'anno seguente venne prodotto il primo processore, l'ARM2 che si presentava con bus dati a 32 bit e 26 bit per quello degli indirizzi.

La comprensione della tipologia di processori RISC è alla base del crescente interesse verso questa nuova tipologia di minicomputer rispetto ai classici. Questa classe di Reduced Instruction Set Computer spiega con il nome stesso la differenza principale identificando un architettura del microprocessore lineare e semplice rispetto ai classici x64 e x86, basati a loro volta su una architettura CISC che utilizza molte istruzioni complesse. Un processore ARM può eseguire ugualmente operazioni complesse ma per far ciò necessita di suddividerla in tante piccole operazioni più semplice rispetto a una cpu x64 in grado di eseguire in una sola operazione un calcolo che richiede una potenza computazionale elevata. Apparentemente potrebbe sembrare da un aspetto negativo dei processori ARM, ma in realtà ci sono dei punti di forza che stanno portando ad una sempre più ampia adozione di questa tipologia di microprocessori che andiamo subito ad analizzare: (19)

- **Risparmio energetico:** La capacità di svolgere piccole operazioni semplici porta ad una riduzione del surriscaldamento e dei consumi. Per questo motivo l'adozione di queste cpu per i dispositivi mobili è incrementata nell'ultimo decennio; infatti, l'autonomia del device è prioritaria rispetto a tutti gli altri aspetti;

- **Flessibilità Open:** Rispetto ai marchi Intel ed AMD, l'architettura ARM è di tipo open. Di conseguenza qualsiasi azienda che ha intenzione di produrre questa cpu può customizzare il chip con caratteristiche che meglio si sposano per l'utilizzo finale e integrando in maniera più rifinita alle necessità del software.

### **3.6.     *Hardware SBC***

Analizzeremo adesso i modelli di questi minicomputer presenti sul mercato al fine di scegliere quello che più si adatta alle nostre esigenze. L'obiettivo è quello di identificare un modello che garantisca un equilibrio tra prezzo e prestazioni.

I modelli di punta dei computer basati su schede single-board sono molteplici e adesso analizzeremo i modelli più utilizzati.

#### **3.6.1. Raspberry Pi 4**

Raspberry Pi 4 è l'ultimo modello prodotto del popolarissimo computer SBC Raspberry Pi. Rispetto alle precedenti generazioni, quali Raspberry B e Raspberry Pi 3, risalta l'incremento incredibile della velocità del processore che contribuisce a rendere molto prestanti le attività multimediali e quelle di connettività. Il modello garantisce la retrocompatibilità con le generazioni precedenti e il consumo energetico risulta pressoché invariato nonostante l'aumento delle prestazioni.

Questo SBC, data la sua prestantza, potrebbe essere paragonabile persino ai sistemi PC x86. Inoltre, è possibile progettare prodotti finali riducendo il tempo per i test di conformità grazie alle certificazioni modulari già attribuite alle schede di comunicazione. (20)

Le sue principali caratteristiche sono:

- Processore: Broadcom BCM2711 Quad-core Cortex-A72 64bit 1.5GHz ARM v8;
- Memoria: 4/8 GB;
- Connettività: 2 x USB 2.0, 2 x USB 3.0, 1 x micro SD, 2 x micro HDMI, 1 x GPIO, 1 x MIPI DPI, 1 x MIPI CSI;
- Comunicazione: Gigabit Ethernet, Bluetooth 5.0, Wifi 802.11b 2,4 GHz/5,0 GHz;
- Alimentazione: 5V DC via USB-C, 5V DC via GPIO.

Prezzo medio: 100€.



### 3.6.2. Odroid-N2

Odroid-N2 presenta il suo punto di forza sulla sua efficienza energetica. Infatti il processore è di alta qualità grazie alla particolare e sofisticata tecnica di fabbricazione a 12nm che lo rende altamente integrato consentendo un risparmio energetico notevole. Sono disponibili due modelli, per quanto riguarda la ram DDR4, uno a 2Gb e l'altro a 4Gb, quindi è ideale per operazioni multitask. Questo SBC viene equipaggiato di default con un dissipatore di calore della cpu posto nella parte inferiore. Tale componente è molto importante per evitare le limitazioni delle prestazioni.

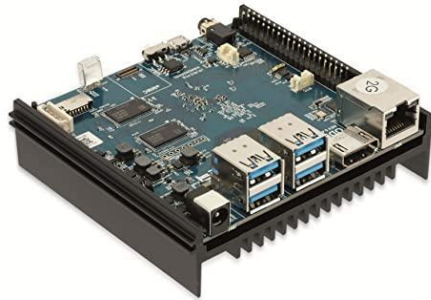
è inoltre possibile eseguire uno streaming video a 4K a 60FPS grazie alla GPU Mali-52. (21)

Le sue principali caratteristiche sono:

- Processore: ARM Amlogic S922X Quad-Core 2x Cortex-A53 a 1.9 Ghz + 2x Cortex-A73 a 1.8 Ghz ;
- Memoria: 2/4 GB
- Connettività: 4 x USB 3.0, 1 x micro SD, 1 x HDMI 2.0, 1 x composito 1 x GPIO, SPDIF ottico, jack audio
- Comunicazione: Gigabit Ethernet LAN (RJ45) Realtek RTL8211F, adattatore WiFi USB opzionale.

- Alimentazione: Jack DC 5.5 mm, adattatore 12V / 2A.

Prezzo medio: 60€.



### 3.6.3. RockPro64

RockPro64 si presenta con un SoC Rockchip RK3399 con core Hexa molto potente. Questo comprende un quad core A53 con clock da 1.4 Ghz e due core A72 da 1,8 Ghz ciascuno. Le specifiche evidenziano immediatamente a colpo d'occhio che questo SBC seppur molto piccolo è estremamente potente. Per quanto riguarda la potenza video è equipaggiata con una GPU Mali T860 MP4 che risulta sufficientemente potente per eseguire contenuti multimediali con una grafica avanzata. Dal punto di vista della memoria ram troviamo una LPDDR4 con 4 canali. Inoltre oltre alla memoria microSD integra uno slot per memorie eMMC. (22)

Le sue principali caratteristiche sono:

- Processore: 4 x ARM Cortex A53 cores @ 1.4GHz, 2 x ARM Cortex A72 cores @ 1.8 GHz ;
- Memoria: 4 GB
- Connettività: 2x USB 2.0 Host, 1x USB 3.0 Host, 1x USB-C Host, 1 x micro SD, 1 x eMMC, 1 x HDMI 4k, 1 x GPIO, jack audio
- Comunicazione: Gigabit Ethernet LAN (RJ45) Realtek RTL8211F;
- Alimentazione: Alimentatore 12V / 2A.

Prezzo medio: 60€.



### 3.6.4. Banana Pi M4

Banana Pi M4 è un computer a scheda singola equipaggiato con una CPU della Realtek modello RTD1395 che presenta un cluster ARM Cortex-A53 64bit quad core. Le prestazioni video sono ottime grazie alla GPU dedicata Mali 470 MP4 che consente un rendering uniforme rendendola una diretta concorrente del Raspberry Pi 4. Di spiacevole sorpresa troviamo un ram aggioranta che non supera i 2Gb LPDDR4, di base 1Gb. Al contrario di altri modelli questa SBC presenta una piccola memoria di massa integrata da 8Gb, comunque espandibile mediante scheda microSD fino a 256Gb. (23)

Le sue principali caratteristiche sono:

- Processore: Realtek RTD1395 quad-core Arm Cortex-A53 ;
- Memoria: 1/2 GB
- Connettività: 4x USB 2.0 Host, 1x USB-C Host, 1 x micro SD, , 1 x HDMI 3.5mm;
- Comunicazione: Ethernet e 802.11, WiFi 5, Bluetooth 4.2;
- Alimentazione: 5 V / 2 A con connettore typoc , supporta PoE.

Prezzo medio: 60€.



### 3.6.5. Asus Tinker Board S

Asus Tinker Board S si presenta con un hardware affidabile e di qualità con una CPU Rockchip RK3288 da 1,8 GHz. La memoria RAM presenta una limitazione, infatti, è soli 2 GB DDR3. Questo SBC integra una GPU Mali - T764 che garantisce ottime capacità di rendering. A differenza degli altri modelli un punto a suo vantaggio è quello di possedere una memoria di massa integrata di ben 16 GB. Per quanto riguarda la connettività notiamo una Gigabit Ethernet, modulo Wifi integrato, modulo Bluetooth integrato, quattro porte esclusivamente USB 2.0, GPIO e HDMI. (24)

Le sue principali caratteristiche sono:

- Processore: Rockchip Quad-Core RK3288 processor 1,8 Ghz;
- Memoria: 2GB Dual Channel DDR3.
- Connettività: 4x USB 2.0 Host, 1x USB-C Host, 1 x GPIO, 1 x HDMI 3.5mm, 6GB eMMC, Micro SD card slot;
- Comunicazione: RTL GB LAN 802.11 b/g/n, Bluetooth V4.0;
- Alimentazione: 1 x Alimentatore scheda Tinker 5V/2-3A.

Prezzo medio: 150€.





### 3.6.6. Aaeon PICO-WHU4

Al contrario dei precedenti questo prodotto è nettamente più potente ed è la soluzione per progetti avanzati. Equipaggiato con un processore Intel Whiskey Lake che collabora con una ram DDR4 che può essere espansa fino a ben 16GB. La grafica UHD Graphics 610/620 offre speciali prestazioni con le due porte HDMI 1.4b. Pur essendo un SBC possiede tutte le interfacce dei classici PC. Troviamo infatti una SATA III compatibile per hard disk o solid state disk. Per quanto riguarda la comunicazione troviamo una Realtek RTL8111G con ben due porte Ethernet, quattro porte tra USB 2.0/3.2. Si evidenzia subito dalle specifiche che questo minicomputer consente qualsiasi tipo di applicazione. Leggermente più grande del raspberry Pi 4 con dimensioni di 100 mm x 72 mm la scheda è utilizzabile anche in condizioni ambientali estreme difficili che gli consentono di operare fino a 60° con una umidità che può raggiungere il 90%. Le sue caratteristiche rendono il prodotto utilizzabile anche in contesti industriali. (25)

Le sue principali caratteristiche sono:

- Processore: 8th Gen Intel® Core™ i7/i5/i3/Celeron Processor SoC;
- Memoria: DDR4 SODIMM Slot x 1, espandibile fino a 32GB
- Connettività: 2 x USB 3.2, 2 x USB 2.0, 2 x Pin header, 2 x RS-232/422/485, 4 x GPIO, 2 x HDMI 1.4b, 1 x SATA 6.0 Gb/s;
- Comunicazione: Realtek 8111 x 2 10/100/1000Mbps;
- Alimentazione: Esclusivamente 12V.

Prezzo medio: 800€.



### **3.7. Software SBC**

Un sistema operativo classico basato su x86 o x64 non può essere eseguito sui SBC a causa dell'architettura differente. Tuttavia, sono stati implementate versioni per architettura ARM. Questi sono sistemi comunque alleggeriti rispetto alle versioni originali come Ubuntu o Kali Linux. Esistono invece delle distro minimali, proprio per permettere una configurazione partendo dal minimo indispensabile.

Analizzeremo adesso alcuni sistemi operativi per architettura ARM compatibili con le macchine analizzate in precedenza.

#### **3.7.1. Flint OS**

Basato su Chromium di Google il sistema operativo Flint OS è molto potente e tutti i servizi e le applicazioni utili sono scaricabili velocemente e direttamente attraverso il repository ufficiale. L'esperienza utente è stata perfezionata grazie al bundle con la piattaforma Chromium. L'avviamento del sistema risulta molto rapido ed efficiente e impiega un tempo minore rispetto ad altri sistemi operativi. Questo sistema consente di creare con un SBC un vero e proprio netbook. Le applicazioni web, differentemente dai tradizionali sistemi operativi, sono alla base di Flint OS quindi la connessione alla rete internet è indispensabile. Direttamente dal Web Store di Chrome è possibile scaricare tutte le app disponibili in base alle proprie esigenze garantendo agli utenti un versatilità di utilizzo elevata e una straordinaria esperienza. (26)

#### **3.7.2. Raspberry Pi OS**

Raspberry Pi OS, chiamato Raspbian nella versione precedente, è uno dei sistemi operativi più utilizzati nei diffusissimi SBC di casa Raspberry. In tutto il mondo milioni di utenti amano questo sistema e lo utilizzano per

realizzare le loro personalizzatissime applicazioni misurate alle proprie esigenze. Questo sistema operativo è realizzato in due versioni: Quella 32bit per garantire la retrocompatibilità con le schede precedenti con architettura a 32bit, e quella a 64bit per offrire sensazionali prestazioni per Raspberry Pi Zero 2, Raspberry Pi 3 e soprattutto ai nuovissimi Raspberry Pi 4. L'applicazione di un modello che utilizza un processore a 64 bit offre diversi vantaggi dal punto di vista tecnico. Nel dettaglio consente una gestione della memoria ram superiore ai 4 GB. Il vero vantaggio è quello di garantire alle app recenti, realizzate esclusivamente per architettura arm di poter essere eseguite, come accade spesso in ambiente linux per closed-source software. (27)

### **3.7.3. DietPi**

DietPi è un sistema altamente ottimizzato che lo rende leggero nel suo utilizzo, dal punto di vista della computazione (utilizzo ram e cpu) richiesta durante la sua esecuzione. Di facile installazione permette una estrema possibilità di personalizzazione in base a qualsiasi tipo di esigenza mediante l'utilizzo della sua interfaccia molto semplice da usare. Una funzionalità molto interessante è quella di poter eseguire il backup del sistema in modo da ripristinarlo ad ogni avvio. Le sue immagini leggere consentono l'esecuzione anche su SBC non specialmente prestanti, compresi i modelli leggermente datati, massimizzando le prestazioni. Come se non bastasse è possibile settare i livelli di priorità dei processi richiesti dalle applicazioni installate tramite un tool chiamato DietPi-Process.

Si conclude dicendo che questo ottimo sistema operativo è configurabile per funzionare in modo personalizzato in base alle esigenze dell'utente. (28)

### 3.7.4. piCore

piCore è il port Raspberry Pi di Tiny Core Linux sviluppato dal Team Tiny Core. È un sistema indipendente progettato da Robert Shingledecker e ora sviluppato da un piccolo gruppo di sviluppatori con un forte supporto della comunità.

piCore Linux non è una distribuzione tradizionale, ma un toolkit per creare sistemi su misura per ogni tipo di esigenza, oltre ad offrire una estrema flessibilità con un ingombro ridotto. Il kernel è molto recente e supporta un set di applicazioni che lo rendono ideale per sistemi personalizzati che permettono di creare vere e proprie apparecchiature per qualsiasi uso. Funziona interamente in RAM senza richiedere installazione convenzionali. Tutte le estensioni montate sono in sola lettura e dopo il riavvio è disponibile lo stesso sistema pulito. L'immagine della scheda SD grezza di base con la versione CLI è di soli 21,5 Mbyte, inclusi il caricatore di avvio RPi, il firmware e i file di supporto. (29)

## 3.8. *Hardware & Software Target*

La scelta di hardware e software è strettamente correlata. Infatti, oltre a essere condizionata dall'equilibrio prezzo/prestazioni si vuole configurare un nuovo sistema operativo forense su processore ARM. Per far ciò è necessario scegliere un sistema operativo progettato e testato su un determinato SBC per stroncare qualsiasi problema di compatibilità limitando qualsiasi tipo di bug che ostacolerebbe il nostro lavoro.

Notiamo un interesse particolare per piCore, il pari della versione base di microCore utilizzata per il progetto della realizzazione di TinyCore Forensic Edition.

Tiny Core FE, come già descritto, è un sistema operativo forense minimale.

Proprio per questo abbiamo scelto piCore per creare un fork del progetto iniziale portando il sistema sui SBC.

Dato che il sistema piCore è stato progettato e testato per Raspberry la scelta dell'hardware è quasi vincolata. Nonostante tutto il Raspberry Pi 4 era comunque candidato a essere scelto come microcomputer per il nostro test, date le elevate prestazioni garantite e il prezzo poco superiore alla media rispetto ai suoi concorrenti.

### **3.9.    *Casi d'uso***

Essendo un hardware assestante e indipendente, la nostra macchina realizzata sarà utile principalmente per eseguire delle acquisizioni statiche. Vale a dire per tutte quelle acquisizioni di memorie che sono spente e disalimentate. Nonostante tutto a seconda del tipo di memoria e di interfaccia che utilizza, distingueremo diversi casi d'uso che adesso andremo ad analizzare.

#### **3.9.1. Acquisizione memorie USB**

Dato che l'interfaccia di comunicazione è unicamente USB non avremo nessuna difficoltà per comunicare con le chiavette di memoria USB. Queste saranno collegate al sistema già avviato per avviare il processo di acquisizione.

Distingueremo le seguenti fasi:

- Copia forense dei dati;
- Hash dei dati;
- Documentare la catena di custodia;
- Analisi dei dati;
- Realizzazione documentazione;

- Analisi dei risultati;
- Redigere conclusione.

### **3.9.2. Acquisizione memorie SD**

Per analizzare delle memorie SD il procedimento è pressoché analogo al precedente, con la sola differenza di utilizzare un adattatore USB. In commercio esistono molti lettori con interfaccia usb utili per i nostri scopi.

### **3.9.3. Acquisizione hard disk**

Nel caso di Hard Disk esterni potremmo ricollegare ai due casi precedenti. La situazione invece cambia nel caso il sequestro riguardi un pc completo, desktop o notebook che sia.

Infatti, se si presume che le prove siano presenti su un pc nella perquisizione si disporrà il sequestro dello stesso. Se al momento del sequestro il pc risulta acceso, sarà preferibile staccare il computer dall'alimentazione elettrica, rimuovendo la spina e l'eventuale batteria. Una volta portato in laboratorio si dovrà eseguire una fase di smontaggio dell'hard disk che sarà collegato come device esterno alla macchina di acquisizione realizzata. Inoltre sarà necessario un adattatore da SATA o IDE (ormai obsoleto) a USB (preferibilmente 3.0).

Distingueremo le seguenti fasi:

- Accesso al sistema;
- Rimozione dell'hard disk;
- Collegamento al sistema forense;
- Memorizzazione prove digitali;
- Hash prove memorizzate;
- Documentazione della catena di custodia;

- Trasporto dei dati in laboratorio;
- Analisi dei dati;
- Realizzazione documentazione;
- Analisi dei risultati;
- Redazione delle conclusioni.

## 4. Realizzazione

### 4.1. *Assemblaggio*

L'hardware minimo per permettere al Raspberry Pi4 di funzionare comprende scheda madre, alimentatore e cavo di connessione video hdmi. Il firmware del dispositivo, in caso di surriscaldamento, limita la potenza di elaborazione della CPU, riducendo le prestazioni, al fine di preservare la scheda madre. L'esigenza di usufruire delle prestazioni massime ha reso necessario anche l'aggiunta della piastra di raffreddamento e di ventole correlate. Infine, è stato utilizzato anche un case che, oltre dal punto di vista estetico, garantisce una protezione dell'hardware. (30)



Le foto riassumono brevemente le semplici fasi dell'assemblaggio della scheda madre con la piastra di raffreddamento e case.

### 4.2. *Installazione piCore*

L'immagine di partenza del sistema operativo “piCore-13.1.0” è stata prelevata dal repository ufficiale:

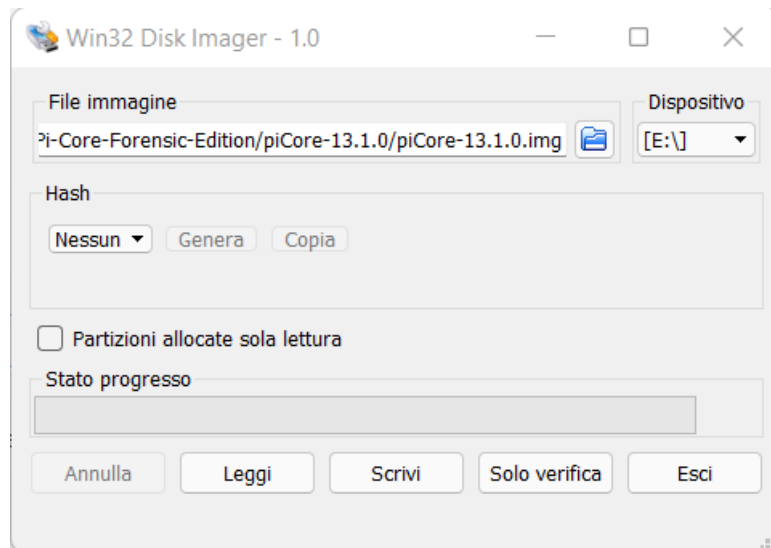
<http://tinycorelinux.net/13.x/armv6/releases/RPi/piCore-13.1.0.zip> (31)



Per eseguire il flash dell'immagine sulla scheda microSD è stato utilizzato il tool "Win32DiskImager":

<https://win32diskimager.org/#download> (32)

Selezionata l'immagine e il dispositivo di destinazione l'operazione è stata avviata con il pulsante scrivi, come mostra la seguente immagine:



Al primo avvio del Raspberry Pi4, con la memoria appena preparata, occorre eseguire un'operazione di ridimensionamento delle partizioni, altrimenti il disco non potrà ospitare nessuno nuovo upgrade, dato che è di dimensioni pari all'immagine scritta.

```

tc@box:~$ sudo fdisk -u /dev/mmcblk0

The number of cylinders for this disk is set to 15279.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
 1) software that runs at boot time (e.g., old versions of LILO)
 2) booting and partitioning software from other OSs
    (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p
Disk /dev/mmcblk0: 15 GB, 16021192704 bytes, 31291392 sectors
15279 cylinders, 64 heads, 32 sectors/track
Units: sectors of 1 * 512 = 512 bytes

Device      Boot StartCHS   EndCHS       StartLBA   EndLBA   Sectors  Size Id Type
/dev/mmcblk0p1  4,0,1      37,63,32     8192      77823    69632 34.0M c Win95 FAT32 (LBA)
/dev/mmcblk0p2  38,0,1     48,63,32    77824     100351    22528 11.0M 83 Linux

Command (m for help): d
Partition number (1-4): 2

Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
p
Partition number (1-4): 2
First sector (32-31291391, default 32): 77824
Last sector or +size or +sizeM or +sizeK (77824-31291391, default 31291391):

```

I comandi da eseguire sono i seguenti:

```

>>sudo fdisk -u /dev/mmcblk0
>>p
>>d
>>2
>>n
>>p
>>2
>>77824 #(valore StartLBA mmcblk0p2p2)
>>ENTER #(ultimo settore default)
>>w
>>sudo reboot
#AL RIAVVIO
>>sudo resize2fs /dev/mmcblk0p2
>>df -h

```

In questo modo abbiamo reso tutta la memoria non allocata disponibile per le future installazioni e aggiornamenti.

piCore, esattamente come TinyCore, è un sistema frugale e la sua esecuzione gira interamente nella memoria ram. Per poter includere qualsiasi tipo di modifica, in modo persistente nel sistema, occorre includere i pacchetti tcz. Molte applicazioni ufficiali, o comunque testate e condivise dalla Community, sono presente nel repository ufficiale (33). Invece per qualsiasi

tipo di personalizzazione è necessario generare un tcz, mediante delle procedure, illustrate in seguito.

Per la maggior parte delle installazioni che eseguiremo utilizzeremo i pacchetti tcz prelevati direttamente da repository ufficiale di PiCore, mediante connessione ad internet del Raspberry con rete cablata Lan. Il comando per scaricare e installare un pacchetto è il seguente:

```
>>tce-load -wi XXX.tcz
```

Invece per includere un tcz in locale nella lista di avvio del sistema operativo, per esempio presente in una memoria usb, come quelli personalizzati che andremo ad implementare, utilizzeremo una funzione differente.

Il comando per installare un pacchetto in locale archiviato su una memoria è il seguente:

```
>>tce-load -i XXX.tcz
```

### **4.3.     *Write Blocker – Blockdev***

La caratteristica principale per convertire il sistema operativo iniziale in modalità forense è quella di garantire con certezza che non venga alterato niente del reperto che si sta analizzando ed inoltre quando si hanno le garanzie relative alla catena di custodia.

Risulta quindi fondamentale garantire l'integrità dei dati contenuti nei dispositivi che verranno montati sul sistema. Il montaggio del file system di un device, in modalità di sola lettura, non garantisce che un driver del kernel non possa scrivere sul dispositivo a blocchi connesso. Questo non significa direttamente che i file interni possono essere compromessi, ma possono essere scritti dei file temporanei, condizione non ammessa comunque dal punto di vista forense. Per questo motivo è necessario l'aiuto di Udev, il device manager del kernel di Linux, che ha il compito di amministrare dinamicamente i dispositivi a blocchi per ogni periferica rilevata dal sistema.

Blockdev è un comando Linux scritto da Andries E. Brouwer e riscritto da Karel Zak e permette di chiamare il block device `ioctl`s dalla linea di comando. Per poter accedere alle sue funzionalità è necessario installare le estensioni `util-linux` mediante i comandi:

```
>>tce-load -wi util-linux.tcz
>>tce-load -wi util-linux-dev.tcz
```

Prendendo spunto dal progetto TinyCore FE andremo a utilizzare la regola `udev forensic.rules` per far eseguire in modo automatico il comando “`blockdev --setro`”, in modo tale da impostare in modalità di blocco scrittura per tutti i dispositivi diversi da *loop* sin dalla fase di avvio del sistema operativo. La regola eseguirà il medesimo comando anche ogni volta che verrà connesso un nuovo device o nel momento in cui avverrà qualche modifica.

La regola `forensic.rule` utilizzata è così composta:

```
ACTION=="add", SUBSYSTEM=="block", KERNEL!="loop*",
RUN+="/usr/local/sbin/blockdev --setro /dev/%k"
ACTION!="remove", SUBSYSTEM=="block", KERNEL!="loop*",
RUN+="/usr/local/sbin/blockdev --setro /dev/%k"
ACTION=="change", SUBSYSTEM=="block", KERNEL!="loop*",
ENV{DISK_RO}=="0", RUN+="/usr/local/sbin/blockdev --setro
/dev/$name"
```

Come già detto Udev è il sottosistema Linux per la gestione degli eventi del dispositivo e la regola che utilizza è costituita da una serie di coppie chiave-valore, separate da virgole.

Le chiavi utilizzate nella `forensic.rule` sono:

- **ACTION** (add, remove, change): Identifica l'azione al verificarsi della quale deve eseguire il comando, ovvero l'inserimento, la rimozione o il cambio di stato di un device;
- **SUBSYSTEM** (block, scsi, pci, usb): Identifica il sottosistema a cui deve essere impartito il comando;

- **KERNEL:** Identifica il tipo di dispositivo su cui eseguire il comando. Nel nostro caso tutti quelli che sono diversi da loop.
- **RUN:** Identifica il comando da eseguire al verificarsi dell'azione, nel nostro caso seguire il comando “blockdev --setro”.

Solitamente, per la maggior parte dei sistemi Linux, è sufficiente inserire la regola nel percorso “/etc/udev/rules.d” con una semplice operazione di copia.

Differentemente, nel sistema frugale che stiamo implementando, per poter includere la regola forense automaticamente, al fine di essere eseguita e rispettata ad ogni avvio è necessario convertirla in un pacchetto tcz.

Per far ciò sono stati eseguiti i seguenti comandi:

```
>>tce-load -wi util-linux.tcz
>>tce-load -wi util-linux-dev.tcz
#creare le seguenti directory
# /tmp/blockdev/etc/udev/rules.d
>>cp /mnt/sda1/forensic.rules
/tmp/blockdev/etc/udev/rules.d
>>mksquashfs blockdev blockdev.tcz
>>mv -v blockdev.tcz /etc/sysconfig/tcedir/optional
>>cd /etc/sysconfig/tcedir
>>echo blockdev.tcz >> onboot.lst
>>sudo reboot
```

Al riavvio del sistema la regola sarà attiva e quindi ogni dispositivo sarà protetto da ogni tipo di scrittura indesiderata garantendo l'immodificabilità del device, a meno che non si voglia abilitare volontariamente la scrittura.

Elenchiamo ora le operazioni principali che possono essere eseguite per utilizzare i vari dispositivi inseriti nel sistema:

- Per consultare lo stato di attivazione del Blockdev sui dispositivi occorre eseguire il seguente comando:

```
>>sudo blockdev --report
```

- Per montare un device in sola lettura occorre eseguire i seguenti comandi (ES. sda1):

```
>>sudo mount -o ro /dev/sda1 /mnt/sda1
```

- Per montare un device in scrittura, prima del montaggio occorre disattivare il blockdev, tramite i seguenti comandi (ES. sda1):

```
>>sudo blockdev --setrw /dev/sda1  
>>sudo mount -o rw /dev/sda1 /mnt/sda1
```

- È possibile riabilitare il blockdev, tramite i seguenti comandi (ES. sda1):

```
>>sudo umount /dev/sda1  
>>sudo blockdev --setro /dev/sda1
```

## 4.4. *Data Dump*

Un comando fondamentale integrato nelle util linux è il dd. Il Data Dump ed è un comando che copia dei dati in blocchi. Permette di clonare bit a bit un disco nella destinazione prescelta, a prescindere dal tipo di filesystem o di sistema operativo e di memorizzare partizioni o interi hard disk. Il comando dd viene utilizzato nei sistemi Unix e Linux per la copia di file di basso livello. Poiché tutto in un sistema simile a Unix è un file, ciò rende dd particolarmente utile per copiare dischi e blocchi di dati specifici da file di grandi dimensioni. L'output del comando è un'immagine raw, che è analizzabile dalla maggior parte degli strumenti software comunemente utilizzati.

Una copia forense di un dispositivo può essere ottenuta tramite tale comando, e grazie al write blocker precedentemente configurato non verrà manomesso alcun singolo bit, mantenendo inalterata la prova originale. (34)

La sintassi del comando dd con i relativi parametri fondamentali sono:

- if: input file o device da clonare (esempio sda);
- of: output file o device su cui clonare o depositare l'immagine;
- bs: block size specifica la dimensione del blocco da copiare in termini di bytes/kilobytes/megabytes e funge come se fosse un buffer di

acquisizione. Questo parametro è molto importante ai fini della velocità. È importante specificare una dimensione non troppo piccola ma neanche troppo grande (consigliato 1M);

- count: numeri di blocchi che si intendono copiare, se non specificati verrà copiato il disco completo.
- oflag: se non specificato la copia del blocco verrà prima appoggiata in memoria e dopo spostata sulla destinazione. È fondamentale settare questo parametro a “direct” per avere ottime prestazioni in modo da riversare direttamente il blocco dalla sorgente alla destinazione.

Per esempio, per clonare un device sda su sdb utilizzeremo il seguente comando:

```
>>dd if=/dev/sda of=/dev/sdb bs=1M oflag=direct
```

Se invece vogliamo creare un’immagine di sda in image.dd il comando sarà:

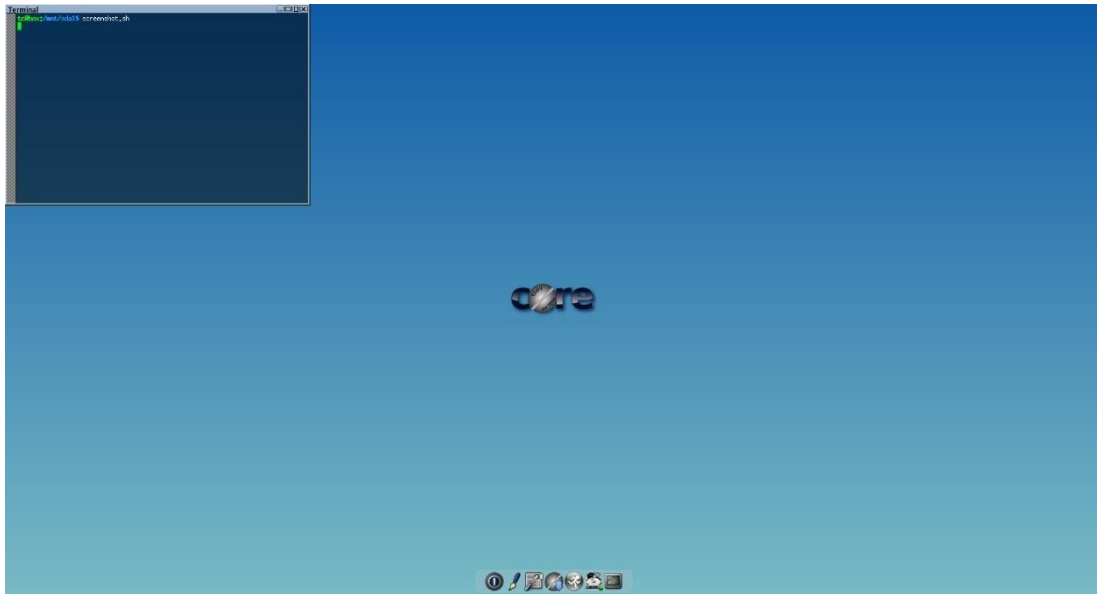
```
>> dd if=/dev/sda of=/mnt/sdb1/image.dd bs=1M oflag=direct
```

## **4.5. Integrazione GUI**

Il sistema PiCore permette l’installazione della GUI con il solo pacchetto “TC”. Dato che la sua dimensione di soli 14Mb è irrisoria rispetto all’usabilità che può garantire una interfaccia grafica si è scelto subito a procedere con questa integrazione. Dopo aver connesso il raspberry Pi4 alla rete internet, mediante rete cablata rj-45, il comando per avviare il processo di installazione è il seguente:

```
>>tce-load -wi TC.tcz  
>>startx
```

Al termine dell’operazione la GUI risulta già avviata e usufruibile.



## 4.6. *Supporto NTFS*

Il sistema piCore supporta di default solo i file system della famiglia FAT, che presenta forti limitazioni. Infatti, la partizione può avere grandezza massima di 8Gb e può ospitare un singolo file di massimo 4Gb. Nell'ottica dell'acquisizione, di fondamentale importanza è la possibilità di poter gestire i file system NTFS. Le estensioni realizzate a tale scopo possono essere integrate mediante i seguenti comandi:

```
>>tce-load -wi ntfs-3g.tcz  
>>tce-load -wi ntfs-3g-dev.tcz
```

Adesso sarà possibile montare in sola lettura una partizione di tipo NTFS mediante i comandi (ES. sda1):

```
>>sudo mount -t ntfs-3g -o ro /dev/sda1 /mnt/sda1
```

Invece, in caso di acquisizione, il drive di destinazione NTFS dovrà essere montato in scrittura nel seguente modo (ES. sda1):

```
>>sudo blockdev --setrw /dev/sda1 /mnt/sda1  
>>sudo mount -t ntfs-3g -o rw /dev/sda1 /mnt/sda1
```



## 4.7. *Compilatore Python*

Python è un linguaggio di programmazione molto versatile. Essendo dinamico e orientato agli oggetti si presta all'uso per svariati tipi di implementazioni software. Un'altra caratteristica fondamentale è quella di offrire supporto stabile all'integrazione con altri linguaggi di programmazione e programmi vari. Inoltre, integra già di default di una vasta libreria standard di facile apprensione e comprensione.

Anche i programmatori che lo utilizzano confermano quanto permetta un effettivo aumento della produttività, inoltre ritengono che consenta loro di implementare codice mantenibile e di superiore qualità.

Proprio per i motivi sopra elencati sono stati integrati nella distribuzione i compilatori python 2.7 e 3.8 per permettere l'eventuale esecuzione di programmi basati su questo linguaggio predominante.

I comandi necessari per installare le due versioni dei compilatori sono i seguenti:

```
>>tce-load -wi python2.7.tcz
>>tce-load -wi python2.7-dev.tcz
>>tce-load -wi python2.7-doc.tcz
>>tce-load -wi python3.8.tcz
>>tce-load -wi python3.8-dev.tcz
>>tce-load -wi python3.8-pip.tcz
>>tce-load -wi python3.8-setuptools.tcz
```

Adesso sarà possibile eseguire direttamente qualsiasi tipo di sorgente mediante il comando:

```
>>python3.8 example.py
```

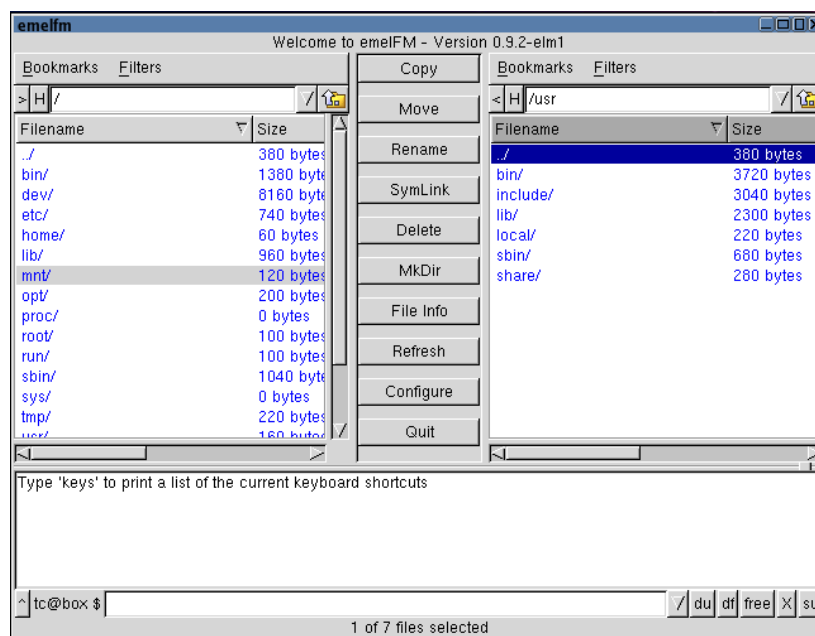
## 4.8. *File Manager*

Integrare un file manager, per poter navigare tra i file presenti nelle memorie in modo pratico, è di fondamentale importanza. Le molteplici utilità

permettono l'esplorazione del contenuto dei drive e l'esecuzione delle operazioni tra i file; nel nostro caso risulta di particolare importanza la copia. Emelfm è un file manager grafico già presente nel repository ufficiale di piCore. Permette di eseguire tutte le operazioni base come la gestione di file e directory, di aprire un percorso direttamente nel terminale e di eseguire i comandi da amministratore. La sua grafica minimale lo rende perfettamente idoneo per le nostre esigenze, soprattutto per non appesantire eccessivamente il sistema.

L'aggiunta del pacchetto all'avvio del sistema necessita l'esecuzione del seguente comando:

```
>>tce-load -wi emelfm.tcz
```

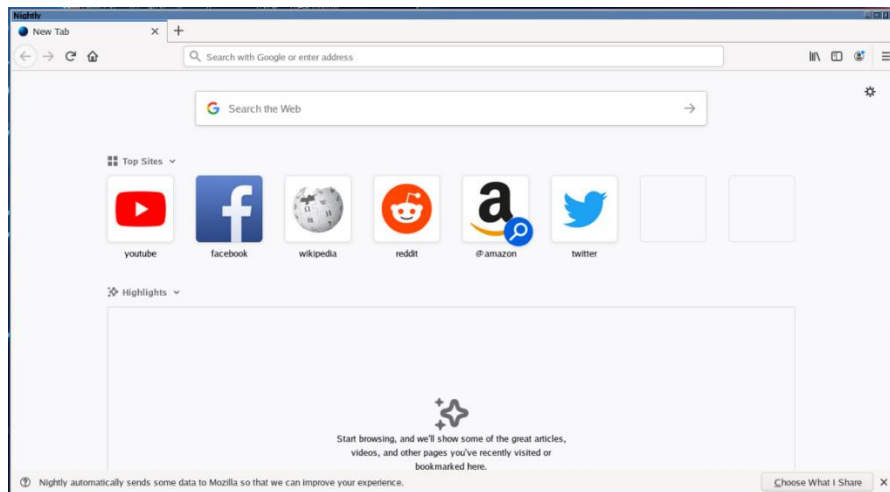


## 4.9. Browser

Dotare il sistema di un browser non è prettamente consigliato, perché il suo utilizzo potrebbe compromettere l'integrità dei dati da acquisire. Tuttavia, potrebbero emergere delle esigenze che renderebbero necessario il suo utilizzo.

Per questa ragione è stato installato il browser firefox tramite il comando:

```
>>tce-load -wi emelfm.tcz
```



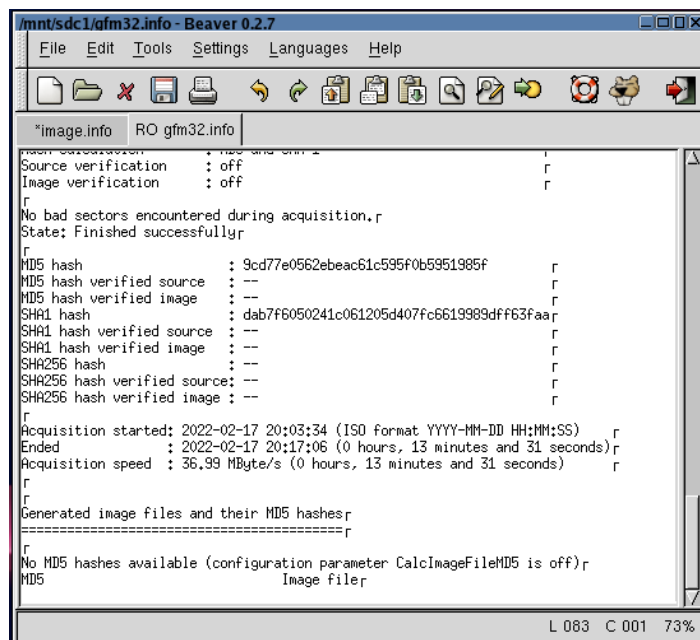
Si raccomanda il suo utilizzo solo in casi di emergenza, e comunque durante le acquisizioni è opportuno non tenere la macchina connessa alla rete internet.

## 4.10. *Text Editor*

Di fondamentale importanza è l'inclusione di un text editor. Questo può essere usato per scrivere una bozza di una relazione o consultare qualsiasi tipo di informazioni.

Il programma Beaver è stato caricato mediante:

```
>>tce-load -wi emelfm.tcz
```



## 4.11. *Ulteriori App*

Dal repository ufficiali sono state incluse anche le seguenti app:

- **wget:** GNU Wget è un pacchetto software gratuito per il recupero di file utilizzando HTTP, HTTPS, FTP e FTPS, i protocolli Internet più utilizzati. È uno strumento a riga di comando non interattivo, quindi può essere facilmente chiamato da script, lavori cron, terminali senza supporto X-Windows; (35)
- **bash:** acronimo di bourne again shell è una shell testuale del progetto GNU usata nei sistemi operativi Unix e Unix-like, come GNU/Linux. Nel sistema macOS era la shell di default fino a macOS Mojave, mentre da macOS Catalina in poi è presente ma non più la predefinita. Bash è disponibile anche per sistemi Microsoft Windows; (36)
- **curl:** viene utilizzato nelle righe di comando o negli script per trasferire i dati. curl è utilizzato anche in automobili, televisori, router, stampanti, apparecchiature audio, telefoni cellulari, tablet, decoder, lettori multimediali ed è il motore di trasferimento Internet per

migliaia di applicazioni software in oltre dieci miliardi di installazioni;  
(37)

- **coreutils:** Le GNU Core Utilities sono le utility di base usate per la manipolazione di file, shell e testo del sistema operativo GNU. Queste sono le utilità di base che dovrebbero esistere su ogni sistema operativo; (38)
- **compiletc:** questa estensione di sviluppo è necessaria per ottenere i componenti essenziali per la compilazione di nuovi pacchetti tcz da generare e includere nel sistema; (39)
- **squashfs-tools:** squashFS è una patch sorgente del kernel Linux che abilita il supporto in lettura di SquashFS nel kernel. Lo strumento mksquashfs crea file system compressi e lo strumento unsquashfs, estrae più file da un file system compresso esistente; (40)
- **vim:** acronimo di "Vi Improved", è un editor di testo. Può essere utilizzato per modificare qualsiasi tipo di testo ed è particolarmente adatto per la modifica di programmi per computer; (41)
- **nmap:** Nmap ("Network Mapper") è un'utilità gratuita e open source che consente il rilevamento della rete e il controllo della sicurezza. E' molto usato in da diversi amministratori di rete e in vari sistemi in quanto risulta utile anche per attività come l'inventario di rete, la gestione dei programmi di aggiornamento dei servizi e il monitoraggio dell'host o del tempo di attività del servizio; (42)
- **xz:** è una nuova utilità di compressione dei dati generica, a riga di comando, simile a gzip e bzip2. Può essere utilizzato per comprimere o decomprimere un file in base alla modalità operativa selezionata. Supporta vari formati per comprimere o decomprimere i file. La selezione di un'utilità di compressione da utilizzare dipenderà principalmente da due fattori, la velocità di compressione e la velocità

di un determinato strumento. A differenza delle sue controparti, xz non è comunemente usato ma offre la migliore compressione. (43)

- **openssh:** è il principale strumento di connettività per l'accesso remoto con il protocollo SSH. Crittografa l'intero traffico al fine di eliminare le intercettazioni, i dirottamenti della connessione e altri attacchi. Inoltre, OpenSSH fornisce un'ampia suite di funzionalità di tunneling sicuro, diversi metodi di autenticazione e sofisticate opzioni di configurazione; (44)
- **usbutils:** Il pacchetto USB Utils contiene utilità utilizzate per visualizzare informazioni sui bus USB nel sistema e sui dispositivi ad essi collegati; (45)
- **linux-5.10.y\_api\_headers:** Il kernel Linux deve esporre un'API (Application Programming Interface) per la libreria C del sistema (Glibc in LFS) da utilizzare. Questo viene fatto ripulendo i vari file dall'intestazione C forniti nel tarball dei sorgenti del kernel Linux. Per compilare i moduli del kernel, sono richiesti gli header del kernel. (46)

Nel dettaglio la sequenza dei comandi eseguiti è:

```
>>tce-load -wi wget.tcz
>>tce-load -wi bash.tcz
>>tce-load -wi bash-dev.tcz
>>tce-load -wi curl.tcz
>>tce-load -wi curl-dev.tcz
>>tce-load -wi coreutils.tcz
>>tce-load -wi compiletc.tcz
>>tce-load -wi squashfs-tools.tcz
>>tce-load -wi vim.tcz
>>tce-load -wi nmap.tcz
>>tce-load -wi xz.tcz
>>tce-load -wi openssh.tcz
>>tce-load -wi usbutils.tcz
>>tce-load -wi linux-5.10.y_api_headers.tcz
```

## 4.12. *RHash*

Recursive Hasher è un utility fondamentale per sigillare digitalmente le acquisizioni è RHash. Essa permette il calcolo e la verifica dei valori hash dei file e supporta gli algoritmi Tiger, BitTorrent BTIH, MD4, MD5, CRC32, SHA1, SHA256, SHA512, ED2K, DC ++ TTH, AICH, RIPEMD-160, GOST R 34.11-94, HAS-160, Whirlpool, Snefru -128/256 e EDON-R 256/512. (47)

RHash permette di creare e verificare Magnet links e eDonkey ed2k:// links rilasciando un output in un formato predefinito (SFV, BSD) o scelto dall'utente. Possiede capacità di elaborare le directory in modo ricorsivo e calcola diverse somme hash in un unico passaggio.

Il programma lavora alla stessa maniera sia sotto Linux, che su Windows o su di un qualsiasi altro sistema operativo.

L'applicazione viene fornita direttamente in codice sorgente C puro, con licenza open source; quindi, per l'installazione occorre compilarla su piCore e successivamente creare il tcz compatibile con il sistema operativo ARM.

La sintassi dei comandi per eseguire tale operazione è la seguente:

```
#COMPILAZIONE TCZ
>>tce-load -wi compiletc.tcz
#estrarre RHash-master.zip su key usb
#montare key usb su picore
>>sudo cp -r /mnt/sdal/RHash-master /tmp
>>cd /tmp/RHash-master
>>sudo ./configure
>>sudo make
>>sudo make install
#creare le seguenti directory
#/tmp/RHash/usr/lib
#/tmp/RHash/usr/local/lib
#/tmp/RHash/usr/local/bin
>>sudo cp /usr/lib/librhash.so.0 /tmp/RHash/usr/local/lib
>>sudo cp /usr/local/lib/librhash.so.0 /tmp/RHash/usr/lib
>>sudo cp /usr/local/lib/librhash.so.0
/tmp/RHash/usr/local/lib
>>sudo cp /usr/local/bin/rhash /tmp/RHash/usr/local/bin
```

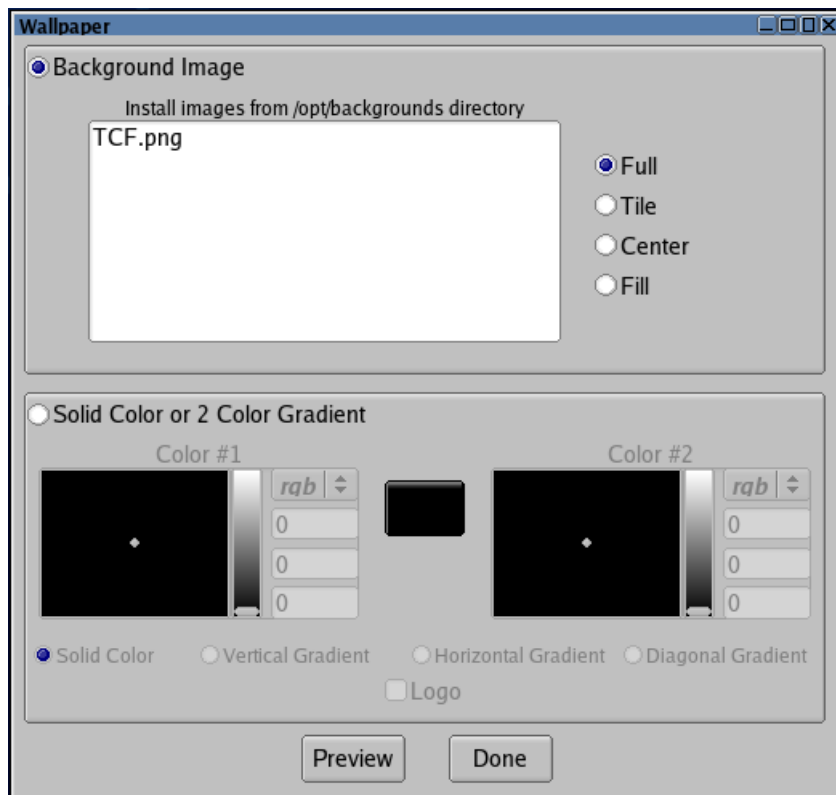
```
>>cd /tmp
>>mksquasfs RHash RHash.tcz
>>sudo cp RHash.tcz /mnt/sda1

#INSTALLAZIONE ONBOOT
>>cp /mnt/sda1/RHash.tcz/tmp
>>cd /tmp
>>mv -v RHash.tcz /etc/sysconfig/tcedir/optional
>>cd /etc/sysconfig/tcedir
>>echo RHash.tcz >> onboot.lst
>>tce-load -i RHash.tcz
```

## 4.13. *Wallpaper*

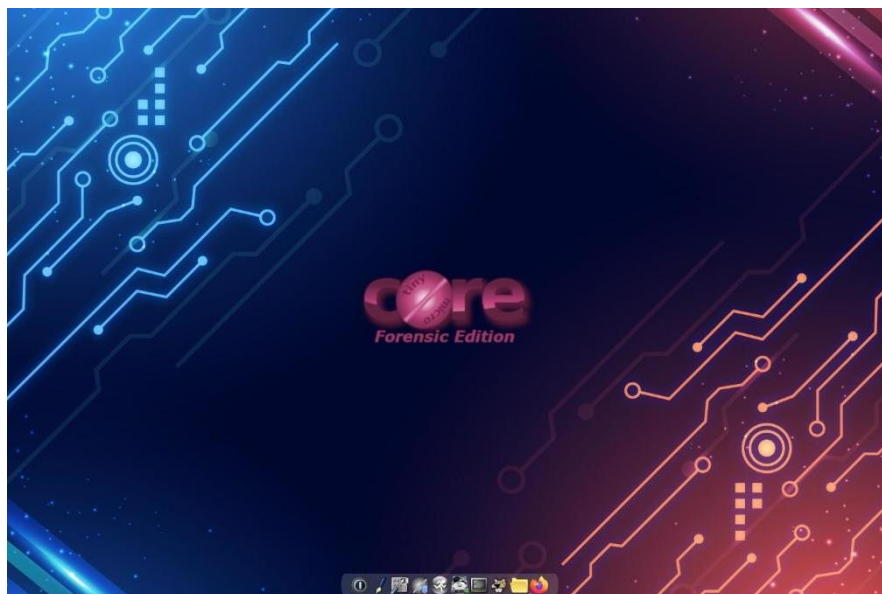
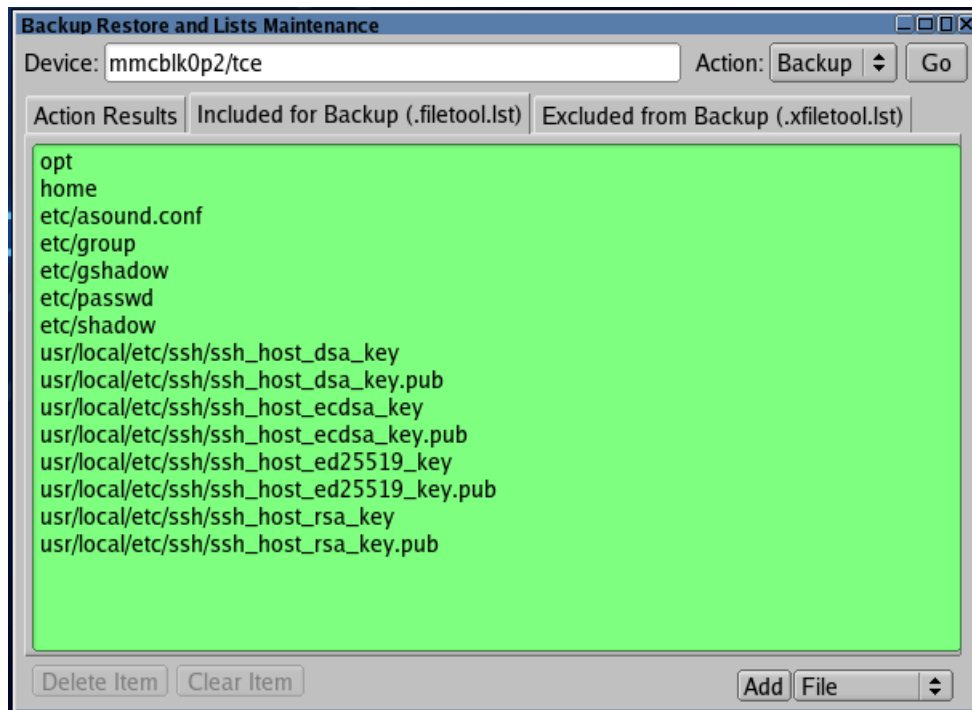
```
>>sudo cp /mnt/sda1/TCF.pn /opt/backgrounds
```

Dal control panel accedere alla sezione wallpaper



Per rendere persistente lo sfondo occorre eseguire il backup



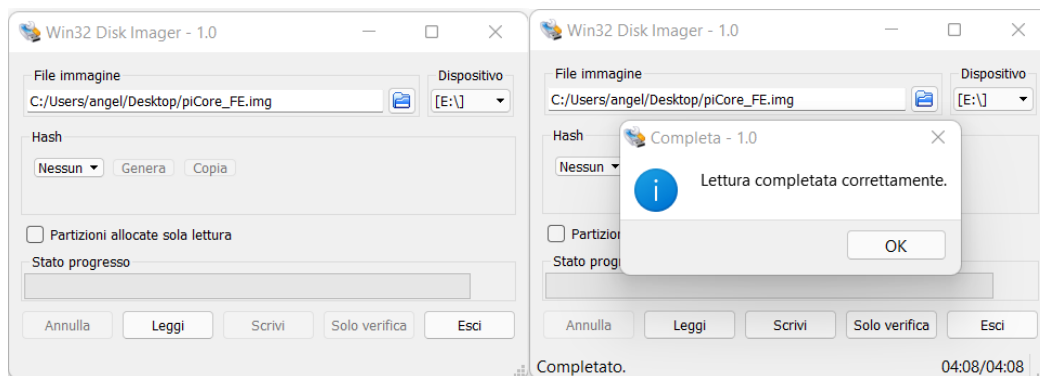


## 4.14. *Script PiCore FE Install*

È stato realizzato lo script `./pife` che automaticamente permette, a partire dalla versione base di pi core, di convertirla in piCore FE. Il comando per eseguire lo script è il seguente:

```
>>sudo mount -o rw /dev/sda1 /mnt/sda1  
>>cd /mnt/sda1/script  
>>./pife.sh
```

## 4.15. *Remaster IMG*



L'immagine ottenuta del nostro sistema configurato è stata rinominata in "pi.Core.Forensic.Edition-v1.0.img".

## 5. Test Acquisizione

Vogliamo ora testare la macchina realizzata per valutarne le performance. Dopo aver definito una serie di memorie di massa, che dovranno rappresentare l'insieme di quelle più utilizzate, andremo ad acquisirle su due configurazioni. Per la prima utilizzeremo un sistema classico e molto diffuso che consiste nell'utilizzare un PC performante con un sistema operativo forense. Con la seconda metteremo alla prova il nostro Raspberry Pi 4 equipaggiato con il sistema operativo PiCore FE realizzato.

### 5.1. Configurazioni

Vengono ora elencate nel dettaglio le specifiche hardware e software delle configurazioni utilizzate per eseguire i test di acquisizione.

#### 5.1.1. Configurazione C1 – Caine/PC

##### **HARDWARE (48):**

- Numero di prodotto: 5SV96EA
- Nome del prodotto: HP Notebook 15-da1000nl
- Microprocessore: Intel® Core™ i7-8565U (frequenza di base 1,8 GHz, fino a 4,6 GHz con tecnologia Intel® Turbo Boost, 8 MB di cache, 4 core)
- Memoria: 8 GB di SDRAM DDR4-2400 (1 x 8 GB)
- Scheda video: NVIDIA® GeForce® MX130 (2 GB di GDDR5 dedicati)
- Unità disco rigido: SATA da 1 TB (5400 rpm), 256 GB PCIe® NVMe™ M.2 SSD
- Display: Schermo FHD SVA antiriflesso con retroilluminazione WLED e diagonale da 39,6 cm (15,6") (1920 x 1080)
- Dispositivo di puntamento: TouchPad con supporto gesti multi-touch
- Connessione wireless: Combo Realtek 802.11b/g/n/a/c (2x2) e Bluetooth® 4.2
- Interfaccia di rete: LAN integrata 10/100/1000 GbE
- Slot di espansione: 1 lettore di schede di memoria multiformato SD

- Porte esterne: 2 USB 3.1 Gen 1 (solo trasferimento dati); 1 USB 2.0; 1 HDMI 1.4b; 1 RJ-45; 1 combo cuffia/microfono
- Tipo di alimentatore: Adattatore di alimentazione CA 65 W
- Tipo di batteria: Ioni di litio a 3 celle, 41 Wh
- Webcam: Fotocamera HP TrueVision HD con microfono digitale integrato
- Caratteristiche audio: Due altoparlanti

## **SOFTWARE (49):**

- Caine 11 64bit

<https://mirror.parrotsec.org/mirrors/parrot/iso/caine/caine11.0.iso>

## **5.1.2. Configurazione C2 – PicoreFE/RaspberryPi4**

## **HARDWARE (50):**

- Processore: Broadcom BCM2711 Quad-core Cortex-A72 64bit 1.5GHz ARM v8;
- Memoria: 4/8 GB;
- Connettività: 2 x USB 2.0, 2 x USB 3.0, 1 x micro SD, 2 x micro HDMI, 1 x GPIO, 1 x MIPI DPI, 1 x MIPI CSI;
- Comunicazione: Gigabit Ethernet, Bluetooth 5.0, Wifi 802.11b 2,4 GHz/5,0 GHz;
- Alimentazione: 5V DC via USB-C, 5V DC via GPIO.
- Temperatura: 0-50°C

## **SOFTWARE:**

- PiCore Foresic Edition

## **5.2. *Device Sorgenti***

Le memorie da acquisire saranno le seguenti:

1. SD SanDisk Ultra XC (64GB);
2. microSD SanDisk Ultra XC I A1 (128GB);
3. KEY USB Kingston DataTravel\_G2 (4GB);
4. KEY USB Kingston DataTravel\_3.0 (32GB);

5. HDD 2.5 WDC WD5000BEVT-22A0RT0 (500GB);
6. HDD 3.5 WDC WD5003ABYZ-011FA0 (500GB);

### 5.3. *Device Destinazione*

Per eliminare il collo di bottiglia in uscita è stato utilizzato un hard disk con box originale di fabbricazione USB3.0. il modello utilizzato è il seguente:

7. TOSHIBA EXTERNAL USB 3.0 DEVICE DTB420 (2000GB)



Come si nota dalle foto, verranno utilizzati anche degli adattatori con interfaccia finale USB, per poter collegare le interfacce non native. Per esempio, ormai essendo obsoleto il vecchio standard IDE, tutti gli Hard Disk utilizzano una porta SATA (esclusi gli ultimi modelli SSD M2). Gli adattatori ci permetteranno la comunicazione fisica tra una porta nativa diversa da quella USB. Un altro esempio sono le memorie SD che necessitano di un lettore di schede USB.

## 5.4. *Acquisizione C1 - Caine/PC*

Questo sistema collaudato e avanzato permette l'acquisizione delle memorie di massa mediante una interfaccia grafica sia per il montaggio che per la fase di acquisizione.

### 5.4.1. Avvio Caine

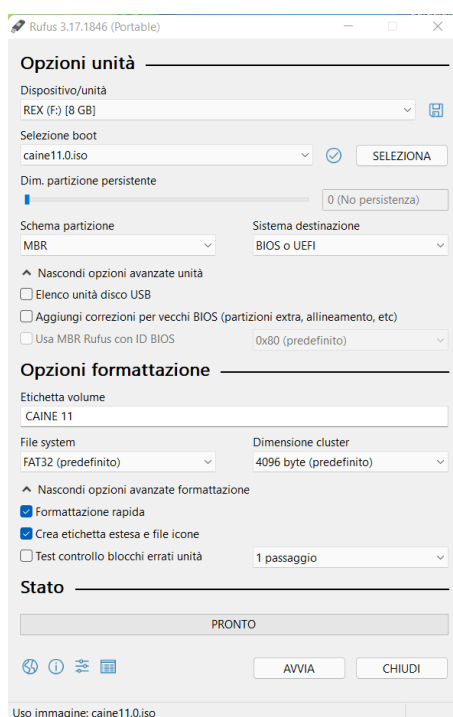
Per prima cosa occorre prelevare la ISO dal repository ufficiale scaricata presso:

<https://www.caine-live.net/page5/page5.html> (51)

Successivamente è necessario creare un supporto avviabile, e per far ciò abbiamo utilizzato il tool rufus scaricato presso:

<https://rufus.ie/it/> (52)

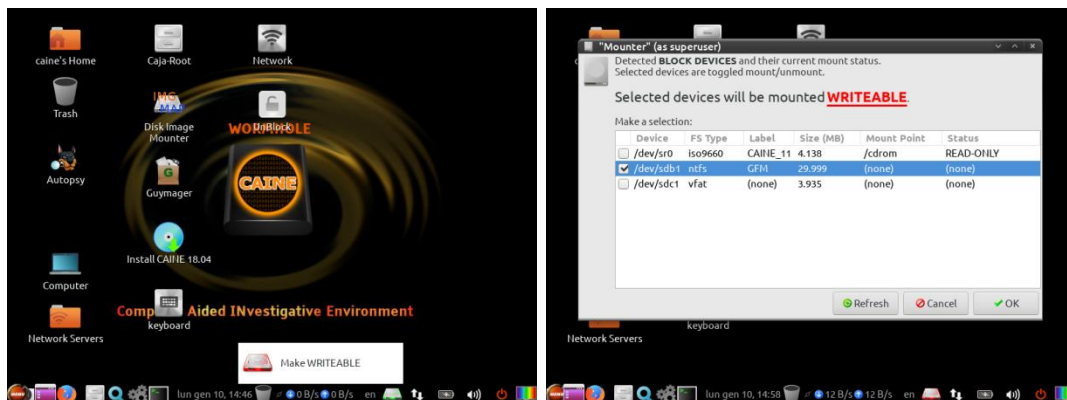
Una volta avviata l'applicazione, intuitivamente abbiamo selezionato l'immagine sorgente scaricata e il drive di destinazione, nel nostro caso una memoria usb da 8GB.



Infine, cliccando sul pulsante avvio abbiamo creato il nostro supporto avviabile in pochi minuti. Riavviato il computer abbiamo selezionato dal bios il boot dalla chiavetta appena create per avviare il sistema operativo forense Caine 11.

### 5.4.2. Abilitazione scrittura e montaggio

Caine non monta automaticamente nessun device e per farlo ci viene in aiuto un tool grafico chiamato mounter. Questo è contrassegnato con un'icona, che mostra un hard disk, (inizialmente verde per indicare il blocco in scrittura attivo) posta al centro della barra delle applicazioni della gui del sistema operativo. Cliccando su di essa con il pulsante destro del mouse, viene mostrata una finestra per disabilitare il blocco in scrittura.



Come mostrato nella seconda immagine il tool mounter permette di selezionare il drive di destinazione dove verranno salvate le acquisizioni. Nella schermata vengono mostrate anche le informazioni relative ai dispositivi come l'etichetta del volume, la dimensione e lo stato di montaggio. Il supporto da acquisire non necessita di alcun tipo di montaggio.

### 5.4.3. Acquisizione

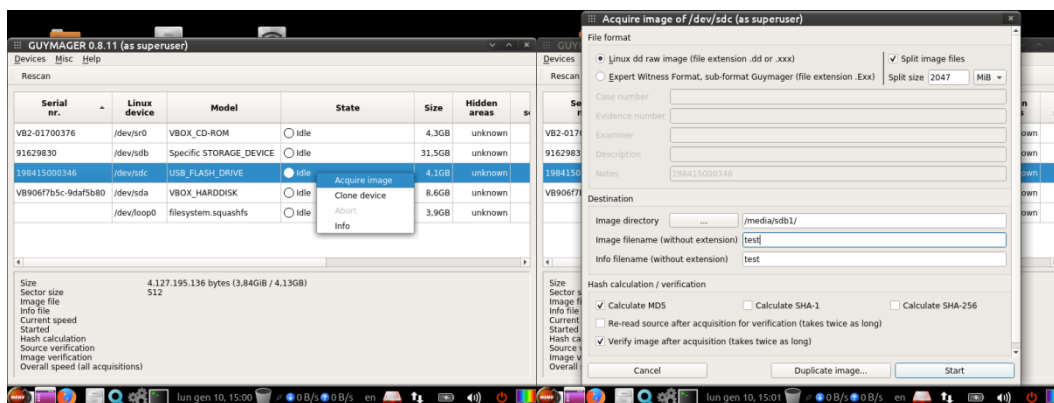
Per la fase di acquisizione viene utilizzata l'applicazione Guymager, già preinstallata in Caine. Nella schermata principale viene mostrata una sorta di tabella che elenca tutte le memorie collegate fisicamente al sistema, anche

se non montate. Vengono mostrate anche informazioni dettagliate come il seriale, il modello e la dimensione.

Dopo aver cliccato sulla memoria da acquisire viene mostrata una schermata dove è possibile selezionare varie opzioni.

Per rendere confrontabile e compatibile il risultato con quello del nostro caso di studi, verrà selezionato quindi il formato linux dd.

Lo split e la verifica della immagine verrà disattivato. Definito il nome e il percorso della acquisizione è possibile avviarla tramite il comando start



#### 5.4.4. Hash dell'immagine

Il tool guymager crea nello stesso percorso un file .info contenente i valori hash calcolati, oltre ad altre informazioni, come per esempio la velocità di acquisizione.

Di seguito un estratto di un file esempio:

```
Acquisition
=====
```

```
Linux device           : /dev/sdb
Device size            : 500107862016 (500,1GB)
Format                 : Linux dd raw image - file
extension is .dd
Image path and file name: /media/sdc1/500wd35.dd
Info path and file name: /media/sdc1/500wd35.info
Hash calculation       : MD5, SHA-1 and SHA-256
Source verification    : off
Image verification     : off
```



No bad sectors encountered during acquisition.  
State: Finished successfully

MD5 hash :  
38aa963592934d4c3946ea0e7894d79a  
MD5 hash verified source : --  
MD5 hash verified image : --  
SHA1 hash :  
b3db286cc30b2bfdb8a21eeae0a11f75a607a7ef  
SHA1 hash verified source : --  
SHA1 hash verified image : --  
SHA256 hash :  
e89321d34aff691e2467f00b8b1e1a186a393bd61a50d8a5e148e2ce2a  
af3144  
SHA256 hash verified source: --  
SHA256 hash verified image : --

Acquisition started: 2022-03-01 22:00:07 (ISO format YYYY-MM-DD HH:MM:SS)  
Ended : 2022-03-01 23:16:47 (1 hours, 16 minutes and 39 seconds)  
Acquisition speed : 103.71 MByte/s (1 hours, 16 minutes and 39 seconds)

## **5.5. *Acquisizione C2 - PicoreFE/RaspberryPi4***

PiCore FE, come già illustrato nei capitoli precedenti, oltre a non montare automaticamente nessuna unità inserita, utilizza un write blocker a basso livello software, per non permettere nessuna alterazione della memoria. Per poter eseguire l'acquisizione di un dispositivo occorrerà abilitare necessariamente in scrittura il drive di destinazione, dove verrà archiviata l'immagine creata e solo successivamente si potrà avviare l'acquisizione della memoria.

### **5.5.1. Identificazione supporto di destinazione**

Quindi, subito dopo aver avviato la macchina, possiamo inserire il nostro supporto di archiviazione che potrà essere facilmente identificato mediante due metodi.

Il primo metodo consiste nell'usare lsblk, comando linux che elenca le informazioni su tutti o sui dispositivi a blocchi specificati. Interroga il file system virtuale / sys per ottenere le informazioni visualizzate. Il comando visualizza i dettagli su tutti i dispositivi a blocchi esclusi tranne i dischi RAM in un formato ad albero per impostazione predefinita. (53)

La sintassi del comando è la seguente:

```
>>lsblk
```

```
loop169 7:169 0 40K 0 loop /tmp/tcloop/libeudev
loop170 7:170 0 180K 0 loop /tmp/tcloop/fontconfig
loop171 7:171 0 476K 0 loop /tmp/tcloop/harfbuzz
loop172 7:172 0 16K 0 loop /tmp/tcloop/libXrender
loop173 7:173 0 8K 0 loop /tmp/tcloop/libXfixes
loop174 7:174 0 4K 0 loop /tmp/tcloop/libXau
loop175 7:175 0 8K 0 loop /tmp/tcloop/libXdmcp
loop176 7:176 0 556K 0 loop /tmp/tcloop/libxml2
loop177 7:177 0 104K 0 loop /tmp/tcloop/at-spi2-core
loop178 7:178 0 48K 0 loop /tmp/tcloop/atk
loop179 7:179 0 24K 0 loop /tmp/tcloop/fribidi
loop180 7:180 0 984K 0 loop /tmp/tcloop/cairo
loop181 7:181 0 16K 0 loop /tmp/tcloop/libpciaccess
loop182 7:182 0 108K 0 loop /tmp/tcloop/wayland
loop183 7:183 0 108K 0 loop /tmp/tcloop/libEGL
loop184 7:184 0 20K 0 loop /tmp/tcloop/libGL
loop185 7:185 0 68K 0 loop /tmp/tcloop/graphite2
sda      8:0    1 29.3G 1 disk
├─sda1   8:1    1 29.3G 1 part
mmcblk0 179:0    0 3.7G 1 disk
├─mmcblk0p1 179:1 0 64M 1 part
├─mmcblk0p2 179:2 0 3.6G 1 part /mnt/mmcblk0p2
zram0    254:0 0 1.9G 1 disk [SWAP]
```

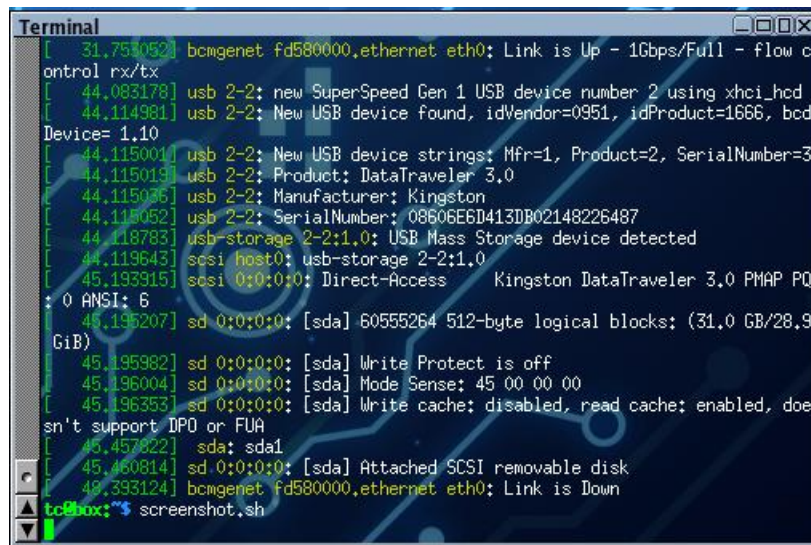
Il secondo metodo consiste nel visualizzare il report del blockdev che Stampa un rapporto per tutti i dispositivi di memorizzazione rilevati dal sistema operativo. Inoltre viene mostrato anche lo stato di abilitazione del write blocker in scrittura per ogni memoria. La sintassi del comando è la seguente:

```
>>sudo blockdev --report
```

```
rw 256 512 4096 0 4096 /dev/loop165
rw 256 512 4096 0 8192 /dev/loop166
rw 256 512 4096 0 12288 /dev/loop167
rw 256 512 4096 0 12288 /dev/loop168
rw 256 512 4096 0 40960 /dev/loop169
rw 256 512 4096 0 163840 /dev/loop170
rw 256 512 4096 0 487424 /dev/loop171
rw 256 512 4096 0 16384 /dev/loop172
rw 256 512 4096 0 8192 /dev/loop173
rw 256 512 4096 0 4096 /dev/loop174
rw 256 512 4096 0 8192 /dev/loop175
rw 256 512 4096 0 569344 /dev/loop176
rw 256 512 4096 0 106496 /dev/loop177
rw 256 512 4096 0 49152 /dev/loop178
rw 256 512 4096 0 24576 /dev/loop179
rw 256 512 4096 0 598016 /dev/loop180
rw 256 512 4096 0 16384 /dev/loop181
rw 256 512 4096 0 110592 /dev/loop182
rw 256 512 4096 0 110592 /dev/loop183
rw 256 512 4096 0 20480 /dev/loop184
rw 256 512 4096 0 69632 /dev/loop185
ro 256 512 4096 0 31457280000 /dev/sda
ro 256 512 4096 2048 31456231424 /dev/sda1
```

Un altro comando interessante è `dmesg` che ci consente di conoscere le caratteristiche hardware delle periferiche riconosciute dal kernel con le relative interfacce supportate.

>>`dmesg`



```
Terminal
[ 31.754052] bcmgenet fd580000,ethernet eth0: Link is Up - 1Gbps/Full - flow c
ontrol rx/tx
[ 44.083178] usb 2-2: new SuperSpeed Gen 1 USB device number 2 using xhci_hcd
[ 44.114981] usb 2-2: New USB device found, idVendor=0951, idProduct=1666, bcd
Device= 1.10
[ 44.115001] usb 2-2: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 44.115019] usb 2-2: Product: DataTraveler 3.0
[ 44.115036] usb 2-2: Manufacturer: Kingston
[ 44.115052] usb 2-2: SerialNumber: 08606E6D413DB02148226487
[ 44.118783] usb-storage 2-2:1.0: USB Mass Storage device detected
[ 44.119643] scsi host0: usb-storage 2-2:1.0
[ 45.193315] scsi 0:0:0:0: Direct-Access Kingston DataTraveler 3.0 PMAP PQ
: 0 ANSI: 6
[ 45.195207] sd 0:0:0:0: [sda] 60555264 512-byte logical blocks: (31.0 GB/28.9
GiB)
[ 45.195982] sd 0:0:0:0: [sda] Write Protect is off
[ 45.196004] sd 0:0:0:0: [sda] Mode Sense: 45 00 00 00
[ 45.196353] sd 0:0:0:0: [sda] Write cache: disabled, read cache: enabled, doe
sn't support DPO or FUA
[ 45.457822] sda: sda1
[ 45.460814] sd 0:0:0:0: [sda] Attached SCSI removable disk
[ 49.393124] bcmgenet fd580000,ethernet eth0: Link is Down
tc@box:~$ screenshot.sh
```

### 5.5.2. Abilitazione scrittura e montaggio

Identificato il drive (nel nostro caso esempio `sda1`) si potrà disabilitare il write blocker mediante il comando:

```
>>sudo blockdev --setrw /dev/sda1 /mnt/sda1
```

Successivamente in caso di partizione di tipo FAT per il montaggio si eseguirà:

```
>>sudo mount -o rw /dev/sda1 /mnt/sda1
```

Invece, nel caso più probabile, di partizione NTFS, la sintassi del comando sarà la seguente:

```
>>sudo mount -t ntfs-3g -o rw /dev/sda1 /mnt/sda1
```

A questo punto il supporto di destinazione è pronto per poter ospitare la creazione dell'immagine di un drive da acquisire.

### 5.5.3. Acquisizione

Dopo aver montato in scrittura il dispositivo di destinazione possiamo inserire nel sistema il drive da acquisire, che non necessita di alcun montaggio nel sistema.

Identificato tale supporto, mediante i comandi già descritti, il sistema è pronto per avviare l'acquisizione. I comandi necessari per far ciò sono i seguenti (nel nostro caso `/mnt/sda1` è il supporto di destinazione e `/dev/sdb` è il supporto sorgente da acquisire):

```
>>cd /mnt/sda1
>>dd if=/dev/sdb of=image.dd bs=10M oflag=direct
```

### 5.5.4. Hash dell'immagine

Il calcolo dell'hash è un'operazione indispensabile per garantire e verificare l'integrità di dati. Infatti, l'immagine acquisita deve essere cristallizzata per evitare qualsiasi alterazione da parte di chiunque. Nel nostro sistema realizzato per calcolare l'hash di un file abbiamo utilizzato il tool RHash tramite i seguenti comandi:

```
//calcolo hash MD5
>>rhash -M image.dd
//calcolo hash SHA1
>>rhash -H image.dd
//calcolo hash SHA256
>>rhash -sha256 image.dd
```

Per creare in un solo passaggio un file di testo contenente i tre checksum calcolati secondo i tre differenti algoritmi hash (MD5 – SHA1 – SHA256) abbiamo utilizzato il seguente comando:

```
>>rhash -M -H -sha256 image.dd > image.info
```

### 5.5.5. Riepilogo acquisizione

La seguente immagine mostra tutte le fasi sopra descritte a partire dalla identificazione dei supporti fino alla sigillatura dell'immagine mediante hash.

```
rw 256 512 4096 0 24576 /dev/loop173
rw 256 512 4096 0 69632 /dev/loop174
rw 256 512 4096 0 16384 /dev/loop175
rw 256 512 4096 0 110592 /dev/loop176
rw 256 512 4096 0 20480 /dev/loop177
ro 131064 512 4096 0 1000204886016 /dev/sda
ro 131064 512 4096 2048 1000202043392 /dev/sda1
ro 256 512 4096 0 31004295168 /dev/sdb
ro 256 512 4096 8064 31000166400 /dev/sdb1
tc@box:~$ sudo blockdev --setrw /dev/sda1
tc@box:~$ sudo mount -t ntfs-3g -o rw /dev/sda1 /mnt/sda1
tc@box:~$ cd /mnt/sda1
tc@box:/mnt/sda1$ dd if=/dev/sdb of=image.dd bs=10M oflag=direct
2956+1 records in
2956+1 records out
31004295168 bytes (31 GB, 29 GiB) copied, 465.149 s, 66.7 MB/s
tc@box:/mnt/sda1$ rhash -M image.dd
146d874ae83d7c4b0ff9589b249c3fff image.dd
tc@box:/mnt/sda1$ rhash -H image.dd
0a2653a9f2365fd62f177a8cd1a60f213d3dfe6f image.dd
tc@box:/mnt/sda1$ rhash --sha256 image.dd
087a8841be0ad0e469187103648ecb51027faaffe6ab9107a5b3bbb5e4a277f9 image.dd
tc@box:/mnt/sda1$ rhash -M -H --sha256 image.dd > image.info
tc@box:/mnt/sda1$
```

## 6. Risultati

### 6.1. *piCore Forensic Edition*

Il file img risultante da questo progetto (pi.Core.Forensic.Edition-v1.0.img 489Mb) è stato caricato su gitHub al seguente link:

<https://github.com/angelotaranto88/piCore-Forensic-Edition-1.0/releases/tag/v1.0> (54)

Al medesimo repository è stato caricato lo script di installazione e le estensioni tcz implementate in questo progetto.

Nel capito 4.2 è già stata illustrata la procedura per l'installazione del sistema su una memoria microSD.

### 6.2. *Test acquisizione*

La seguente tabella mostra i dispositivi selezionati su cui abbiamo eseguito le acquisizioni con le relative caratteristiche tecniche:

DEVICE TIPOLOGI A	DEVICE MODELLO	SIZE DIMENSIONE (GB)	INTERFACCI A NATIVA	INTERFACCIA ADATTATOR E
SD	SanDisk Ultra XC	64	SD	USB 2.0
microSD	SanDisk Ultra XC I A1	128	microSD	USB 2.0
KEY USB	Kingston DataTravel_G2	4	USB 2.0	USB 2.0
KEY USB	Kingston DataTravel_3.0	32	USB 3.0	USB 3.0
HDD 2.5	WDC WD5000BEVT- 22AORT0	500	SATA 2.6	USB 3.0
HDD 3.5	WDC WD5003ABYZ-011FA0	500	SATA 3.0	USB 3.0

I relativi benchmark di acquisizione registrati mediante i test di acquisizione nelle due configurazioni sono i seguenti:

DEVICE TIPOLOGIA	DEVICE MODELLO	SIZE DIMENSIONE (GB)	C1 - CAINE 11 BENCHMARK (Mb/s)	C2 - PI CORE FE 2.2 BENCHMARK (Mb/s)
SD	SanDisk Ultra XC	64	18,96	19,81
microSD	SanDisk Ultra XC I A1	128	18,04	19,13
KEY USB	Kingston DataTravel_G2	4	23,59	24,84
KEY USB	Kingston DataTravel_3.0	32	71,58	73,06
HDD 2.5	WDC WD5000BEVT-22A0RT0	500	58,79	56,60
HDD 3.5	WDC WD5003ABYZ-011FA0	500	103,71	102,73

Come si può notare dalla tabella il Raspberry pi 4 utilizzato abbinato al sistema piCore Fe realizzato ha mantenuto un benchmark pressoché identico a quello della configurazione 1 consentendo ai dispositivi di comunicare alla propria velocità massima in fase di acquisizione.

## 7. Sviluppi Futuri

Dal punto di vista Hardware, per rendere il dispositivo completamente portatile, sarebbe opportuno installare un display touch integrato. In questo modo il nuovo strumento si potrebbe trasportare e utilizzare senza l'aggiunta di hardware aggiuntivo (monitor, mouse) esattamente come i dispositivi realizzati ad hoc analizzati precedentemente.

La seguente immagine mostra un display presente in commercio: (55)



Dal punto di vista software invece, il sistema operativo necessiterebbe di alcune funzionalità avanzate come:

- **Blockdev grafico:** Occorrerebbe realizzare un'interfaccia grafica per gestire al meglio il montaggio dei dispositivi in scrittura (indispensabile come destinazione delle acquisizioni);
- **Imager grafico:** Anche la fase di acquisizione sarebbe molto più semplice e usabile se l'iterazione con l'utente fosse gestita con un'interfaccia grafica. In questa si andrebbero a settare i parametri dell'acquisizione, compresa sorgente e destinazione, per ricavare l'immagine forense;
- **Supporto partizioni mac:** Occorrerebbe un'utility per potere montare e analizzare partizioni APFS (Apple File System) e



Mac OS. Infatti molti dispositivi utilizzati sono partizionati secondo questi schemi non compatibili direttamente con linux.

## 8. Bibliografia

1. **Wikipedia.** [Online] [https://it.wikipedia.org/wiki/Scienza\\_forense](https://it.wikipedia.org/wiki/Scienza_forense).
2. —. [Online] [https://it.wikipedia.org/wiki/Alphonse\\_Bertillon](https://it.wikipedia.org/wiki/Alphonse_Bertillon).
3. —. [Online] [https://en.wikipedia.org/wiki/Edmond\\_Locard](https://en.wikipedia.org/wiki/Edmond_Locard).
4. **wiki.** [Online]  
[https://it.wikipedia.org/wiki/Informatica\\_forense#:~:text=Il%20concetto%20di%20informatica%20forense,e%20riconosciuti%20come%20crimini%20informatici..](https://it.wikipedia.org/wiki/Informatica_forense#:~:text=Il%20concetto%20di%20informatica%20forense,e%20riconosciuti%20come%20crimini%20informatici..)
5. —. [Online] [https://it.wikipedia.org/wiki/Catena\\_di\\_custodia](https://it.wikipedia.org/wiki/Catena_di_custodia).
6. —. [Online] [https://en.wikipedia.org/wiki/Daubert\\_standard](https://en.wikipedia.org/wiki/Daubert_standard).
7. **uniba.** [Online]  
[https://www.uniba.it/ricerca/dipartimenti/informatica/tutorato/orientamento-e-tutorato-1/attivita-di-orientamento-anni-precedenti/orientamento-consapevole-anni-precedenti/orientamento-consapevole-2021/nono\\_seminario/digital-forensics](https://www.uniba.it/ricerca/dipartimenti/informatica/tutorato/orientamento-e-tutorato-1/attivita-di-orientamento-anni-precedenti/orientamento-consapevole-anni-precedenti/orientamento-consapevole-2021/nono_seminario/digital-forensics).
8. **shop, forensic.** [Online] <https://forensicshop.it/home-page/144-tableau-tx1-forensic-imager-kit.html>.
9. **sistemieservizi.** [Online] <https://www.sistemieservizi.net/cybercrime-digital-forensics/news/62/Logicube-Falcon-Neo-/>.
10. **falcon.** [Online] <https://www.logicube.com/shop/forensic-falcon-neo/>.
11. **Atola.** [Online] <https://www.atola.com/products/insight/>.
12. **opentext.** [Online]  
<https://security.opentext.com/tableau/hardware/details/td2u>.
13. **wiki.** [Online] [https://en.wikipedia.org/wiki/CAINE\\_Linux](https://en.wikipedia.org/wiki/CAINE_Linux).

14. **cybersecurity360**. [Online] <https://www.cybersecurity360.it/soluzioni-aziendali/distribuzioni-forensi-per-linux-a-cosa-servono-e-le-migliori-per-le-analisi-forensi/>.
15. **tsurugi**. [Online] <https://tsurugi-linux.org/>.
16. **Ruffo**. [Online] <https://github.com/SaraRuffo/Tiny-Core-Forensic-Edition---Tesi/releases/tag/v1.0>.
17. **Wikipedia**. [Online] [https://it.wikipedia.org/wiki/Legge\\_di\\_Moore](https://it.wikipedia.org/wiki/Legge_di_Moore).
18. —. [Online] [https://it.wikipedia.org/wiki/Single-board\\_computer#:~:text=Un%20Single%2Dboard%20computer%2C%20noto,richieste%20per%20un%20computer%20funzionale..](https://it.wikipedia.org/wiki/Single-board_computer#:~:text=Un%20Single%2Dboard%20computer%2C%20noto,richieste%20per%20un%20computer%20funzionale..)
19. **Bianco, Matteo**. [Online] <https://www.weisoft.it/prodotti/office365/item/processor-arm.html>.
20. **futurashop**. [Online] <https://www.futurashop.it/raspberry-pi-4-tipo-b-con-8gb-di-memoria-7310-rpi4-8gb>.
21. **hardkernel**. [Online] <https://www.hardkernel.com/shop/odroid-n2-with-4gbyte-ram-2/>.
22. **pine64**. [Online] <https://www.pine64.org/rockpro64/>.
23. **wikibanana**. [Online] [https://wiki.banana-pi.org/Banana\\_Pi\\_BPI-M4](https://wiki.banana-pi.org/Banana_Pi_BPI-M4).
24. **asus**. [Online] <https://www.asus.com/it/Networking-IoT-Servers/AIoT-Industrial-Solutions/All-series/Tinker-Board-S/>.
25. **aaeon**. [Online] <https://www.aaeon.com/en/p/pico-itx-boards-pico-whu4>.
26. **flint**. [Online] <https://archiveos.org/flint/>.
27. **raspberrypi**. [Online] <https://www.raspberrypi.com/software/operating-systems/>.
28. **dietpi**. [Online] <https://dietpi.com/>.

29. **tinycorelinux.** [Online]  
<http://tinycorelinux.net/5.x/armv6/releases/README>.
30. **Zzed.** [Online] <https://www.raspberryyitaly.com/ventola-si-ventola-no/>.
31. **piCore.** [Online]  
<http://tinycorelinux.net/13.x/armv6/releases/RPi/piCore-13.1.0.zip>.
32. **win32diskimager.** [Online] <https://win32diskimager.org/#download>.
33. **tinycoretcz.** [Online] <http://tinycorelinux.net/13.x/x86/tcz/>.
34. **unixtutorial.** [Online] <https://www.unixtutorial.org/commands/dd>.
35. **gnu.** [Online] <https://www.gnu.org/software/wget/>.
36. **wiki.** [Online] <https://it.wikipedia.org/wiki/Bash>.
37. **curl.** [Online] <https://curl.se/>.
38. **gnu.** [Online] <https://www.gnu.org/software/coreutils/>.
39. **distro.** [Online] <http://distro.ibiblio.org/tinycorelinux/3.x/tcz/>.
40. **tldp.** [Online] [https://tldp.org/HOWTO/html\\_single/SquashFS-HOWTO/#:~:text=SquashFS%20is%20distributed%20as%20a,an%20existing%20squashed%20file%20system..](https://tldp.org/HOWTO/html_single/SquashFS-HOWTO/#:~:text=SquashFS%20is%20distributed%20as%20a,an%20existing%20squashed%20file%20system..)
41. **computerhope.** [Online]  
<https://www.computerhope.com/unix/vim.htm>.
42. **nmap.** [Online] <https://nmap.org/>.
43. **Linux-Console.** [Online] <https://it.linux-console.net/?p=401>.
44. **openssh.** [Online] <https://www.openssh.com/>.
45. **tinycorelinux.** [Online]  
<http://forum.tinycorelinux.net/index.php?topic=18395.0>.
46. **Llorente, Rubén.** [Online] <https://www.linux-magazine.com/Issues/2021/243/Tiny-Core-Linux>.
47. **giyhub.** [Online] <https://github.com/rhash/RHash>.

48. **15-da1000nl, HP Notebook.** [Online] <https://support.hp.com/it-it/document/c06234615>.
49. **download, Caine11.** [Online] <https://mirror.parrotsec.org/mirrors/parrot/iso/caine/caine11.0.iso>.
50. **LOCATELLI, MARCO.** [Online] <https://leganerd.com/2020/05/28/raspberry-pi-4-arriva-la-versione-da-8gb-prezzo-e-specifiche/>.
51. **cainelive.** [Online] <https://www.caine-live.net/page5/page5.html>.
52. **rufus.** [Online] <https://rufus.ie/it/>.
53. **frameboxxindore.** [Online] <https://frameboxxindore.com/it/linux/how-use-blkid-command-in-linux.html>.
54. **Edition, piCore Forensic.** [Online] <https://github.com/angelotaranto88/piCore-Forensic-Edition-1.0/releases/tag/v1.0>.
55. **Amazon.** [Online] <https://www.amazon.it/touchscreen-Raspberry-custodia-pollici-ventola/dp/B07WSVS1Q1>.