A Case History: the OSINT

Author : Vincenzo Digilio

Date : 30 Ottobre 2019

Avete mai pensato a cosa muove un Hacker?
È semplicemente un reflusso di ormoni adolescenziali oppure vi è qualcosa dietro, **qualcosa di più profondo?** Come l'esigenza e la necessità di reagire ad un sopruso o di liberarsi dalle sbarre dorate di questa società? Due testi su questo argomento la dicono lunga: il primo è *"La Coscienza di un Hacker"*[1]; il secondo *"The Fallen Dreams: Diario di un Hacker"*, reperibile facilmente per una decina di euro.
Entrare nella mente di un Hacker è uno degli aspetti fondamentali per prevenire un attacco, soprattutto perché spesso, quando si tenta di intervenire a posteriori, si rivela essere già troppo tardi.

Ora la domanda è: esistono davvero queste "sbarre dorate"? Vi è realmente una guerra silente in atto che prende piede, giorno dopo giorno, sullo sfondo delle nostre vite?
Analizziamo **alcuni fatti**.
Lo Xinjiang è una regione autonoma della Cina nord-occidantale, che ha più o meno gli stessi abitanti di Pechino. La Cina sta usando questi abitanti come cavie per sperimentare nuove forme di controllo della tecnologia, investendo massicciamente nello sviluppo di IA e Software. I cittadini sono obbligati a scaricare sui loro dispositivi diverse app in grado di analizzare flussi di dati in ingresso e in uscita dai loro smartphone. Tali dati vengono successivamente trasmessi a un server centralizzato, in grado di stabilire un insieme di comportamenti dell'individuo e creare una categoria di azioni ritenute "standard". Ora, se il comportamento di un cittadino dovesse discostarsi da quello "standard", il server avvertirebbe subito dell'anomalia. Oltre a questo, numerose startup cinesi di cyber sicurezza hanno ricevuto ingenti finanziamenti per lo sviluppo di sistemi di riconoscimento facciale, che in Cina si affermano come realtà più che consolidata e risultano ormai un affare globale. Se, a tutto questo, aggiungiamo droni dal peso di 200 grammi dotati di fotocamera ad alta definizione che simulano la reale aerodinamica degli uccelli, capiamo **l'urgenza di proteggere la nostra libertà**.

Similmente, nel nostro Paese, vi è una profanazione costante della privacy: veniamo profilati in maniera sistematica quando navighiamo su internet, quando facciamo la spesa, quando dormiamo, quando guardiamo una pubblicità; viene persino analizzata la nostra risposta facciale a un determinato stimolo al fine di capire se ci piaccia o meno quel determinato

oggetto, oppure se quello che stiamo leggendo è o non è nelle nostre corde.

Le **tracce** che quotidianamente lasciamo sulla rete sono innumerevoli, tra dati testuali, video, audio… non solamente i privati, ma anche molte aziende sottovalutano il potere dello "Spionaggio da Fonti Libere".
L'OSINT, acronimo di **Open Source INTelligence**, si occupa proprio della "raccolta d'informazioni mediante la consultazione di fonti di pubblico accesso"[2].

Recentemente, un'agenzia investigativa ci ha chiesto di collaborare per un caso di diffamazione, riguardante un personaggio molto in vista. Vi confesso che non amo particolarmente far da balia a un paio di damerini quindi, quando il nostro collega mi propose il lavoro, la risposta fu ben più colorita della seguente:

«Se lui non riesce a tenersi stretto il suo profilo, noi non possiamo farci molto».
«Non è solo questo, pare che siano stati coinvolti anche i suoi figli di dodici e sedici anni» replicò il mio socio. Poi, dopo una breve pausa, aggiunse «Le foto dei minori sono apparse anche su alcuni siti anonimi… inoltre, pare che sia presente in rete un elenco con diversi numeri di telefono associati ad altri personaggi».
«Traduci: "anonimi". Tor? L2P? Freenet?[3]» domandai.
Il mio socio scosse la testa: «Internet».
«Cioè, sono sul "Surface Web"? (Il cosiddetto web di superficie, dove quotidianamente navighiamo)» dissi fra lo stupito e l'incredulo.
«Lo accettiamo questo lavoro oppure no?» replicò lui con un sorrisetto sornione.

Ormai aveva stuzzicato la mia curiosità, quindi non rimaneva che approfondire…

PARTE I: THE REVERSE BRUTE FORCE

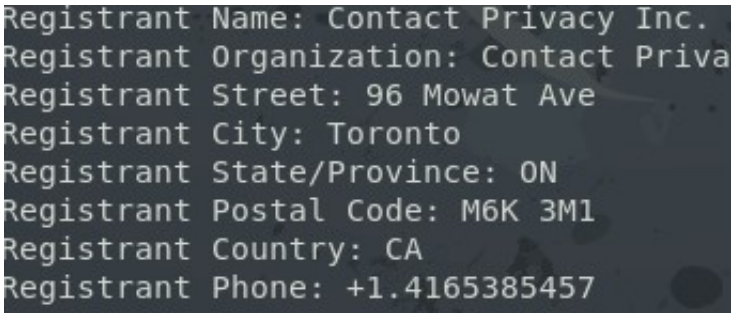
Solitamente, chi è avvezzo a questo genere di cose (un cyber criminale, o un cracker) non utilizza il web di superficie per uplodare contenuti, a meno che non si tratti di un atto di hacktivismo o deliberatamente volto a far emergere fatti o dati all'"opinione pubblica". Anche se la vittima avrebbe potuto potenzialmente rappresentare un bersaglio per gli Hacker, le modalità utilizzate ed il coinvolgimento di minori, li scagionava. Almeno in linea teorica.

Decisi come prima cosa di dare un'occhiata al dominio imputato, attualmente sotto sequestro. Per ovvie ragioni, non potrò parlare direttamente del caso in questione, ma farò un esempio parallelo ad esso:

La prima cosa che feci, fu un *whois* del sito utilizzando la famosa distro: Kali Linux[4].

whois example.net --verbose

Per chi non fosse pratico, *whois* è un protocollo di rete che consente di interrogare il *server database*, per stabilire a quale provider internet appartenga un determinato IP o DNS. Ciò che attirò la mia attenzione furono le informazioni successive mostrate dal comando, in particolare l'intestatario e la data di registrazione.

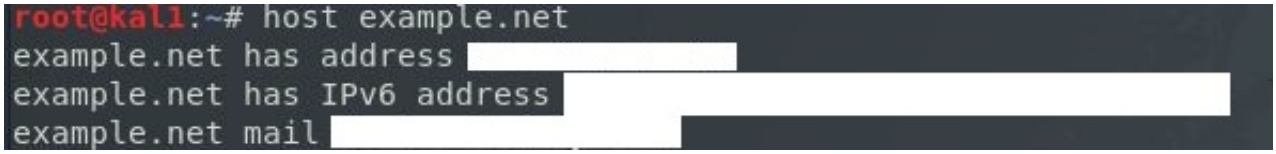


(L'immagine è estratta dall'output del comando a puro scopo esemplificativo)

L'opzione *verbose* mi consentiva di collezionare eventuali informazioni aggiuntive sullo stato dell'output.

Ciò che mi colpì fu il dominio, intestato a una vecchia società (che chiameremo "Example S.p.A**"**). Feci qualche ricerca su internet e trovai diversi articoli che parlavano della chiusura di diverse holding, assorbite da un grosso gruppo finanziario. Un dominio morto, quindi, apparentemente.

Utilizzai il comando: **host** example.net, per assicurarmi dell'IP.



(L'immagine è estratta dall'output del comando a puro scopo esemplificativo)

La domanda che mi posi successivamente fu: esistono altri IP registrati da "Example S.p.A"? Per scoprirlo decisi di fare un Reverse Brute Force degli IP. Mi spiego meglio: avevo ottenuto l'indirizzo del sito internet grazie al comando "host: **194.168.244.97 dominio** example.net". Ora, non mi restava che scoprire se ci fossero altri IP dell'insieme hosts 192.168.244.1-254 registrati a nome della holding "Example S.p.A"… e se sì, quali?

Improvvisai un semplice script in bash[5], del tipo:

```
#!/bin/bash
for ip in $(cat listadegliIP.txt);
  # per ogni "ip"
contenuto nel file di testo den
ominato: "listadegliIP.txt"
do whois $ip | grep "Example S.p
.A";  #esegui il commando "whois" dell'IP
```

```
e stampa a video (tramite il comando grep) solo le stringhe che conten
gono la dicitura "Example S.p.A"  echo $ip;  #mostrami
 l'IP associato  Done  #fine
```

Sorpresa! Altri 3 IP risultavano ancora registrati a nome di questa vecchia holding. Effettuai un altro *whois* esteso per ogni indirizzo IP, ma le informazioni ottenute non differivano dall'originale.

PARTE II: Google Hacking

A questo punto, decisi di utilizzare un'altra tecnica: il *Dorking* (conosciuta anche come Google Hacking). Essa consiste in interrogazioni mirate del web, sfruttando Google. Esiste un database di riferimento con cui sbizzarrirsi: https://www.exploit-db.com/google-hacking-database.

Quindi aprii il browser e utilizzando la barra di ricerca di Google digitai:

- *site:example.net ext:xml* | *ext:conf* | *ext:cnf* | *ext:reg* | *ext:inf* | *ext:rdp* | *ext:cfg* | *ext:txt* | *ext:ora* | *ext:ini*

Dove ***"site"*** è il sito target ed ***"ext"*** sta per "estensione". Ergo, avrei cercato nei tre siti registrati a nome della holding Example S.p.A eventuali file di configurazione esposti. Eseguii una decina di Dork, fra cui "*site:example.net inurl:login*", al fine di cercare le pagine di Login in correlazione con il target.

Ma la query che aggiunse un tassello al nostro puzzle fu:

- *site:example.net ext:doc* | *ext:docx* | *ext:odt* | *ext:pdf* | *ext:rtf* | *ext:sxw* | *ext:psw* | *ext:ppt* | *ext:pptx* | *ext:pps* | *ext:csv*

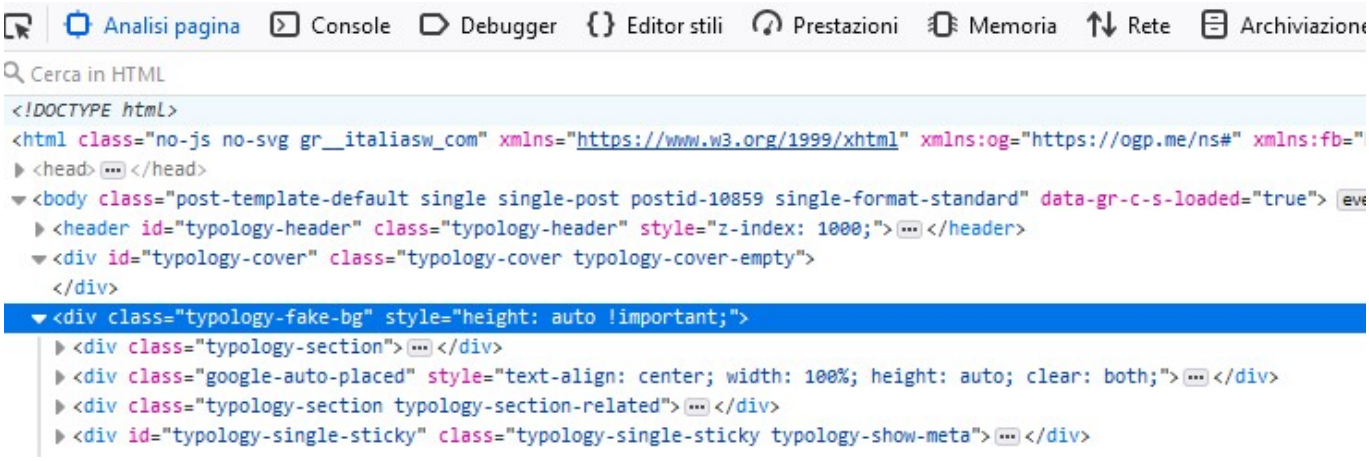
In questo caso, cercai tutti i documenti esposti pubblicamente nei siti target Example S.p.A.

Nel terzo sito, ben annidati nel ginepraio dei sotto URL del dominio, apparvero due pdf, ignaramente esposti sulla rete. Non posso citare cosa contenevano, ma posso dirvi che menzionavano un'altra holding estera, ritenuta responsabile del crollo della società (che chiameremo con molta fantasia "Example2 S.p.A").

PARTE III: RESURRECT

Dalle semplici ricerche su Google della società, non apparve nulla. Molte portavano a vicoli ciechi o pagine rimosse e non più reperibili. Decisi di avvalermi di un'estensione del browser nota come "Resurrect Pages"[6], che permette di raccogliere informazioni da vari aggregatori come Achive.is, Google Cache, WebCite, Google, etc…

A questo punto, tornai su alcune di quelle pagine che mi avevano restituito un errore di Google “404” ed eseguii il “Resurrect”, tramite l’estensione del browser che avevo appena scaricato. Mi apparvero a video alcune vecchie pagine di Example2 S.p.A, incomplete. Bastò poi un’analisi superficiale con un banale F12[7] per rivelare il dominio a cui faceva riferimento la società: **example2.eu**.



(L’immagine è estratta dall’output del comando a puro scopo esemplificativo)

PARTE IV: FORWARD LOOKUP BRUTE FORCE

Decisi che questo dominio appena trovato meritava un’analisi più approfondita, così eseguii quello che in gergo viene chiamato “Forward LookUp BruteForce”. L’idea alla base è quella di fare un *guessing* (tirare ad indovinare) i nomi dei server validi (appartenenti al dato dominio), tendando ogni volta di risolverli.

Quindi produssi un file .txt con un centinaio di hostname più comuni.

```
www
mail
ftp
localhost
webmail
smtp
pop
a
acceptatie
access
accounting
accounts
```

(L’immagine è a puro scopo esemplificativo)

E scrissi il semplice script:

```
#!/bin/bash
for ip in $(cat hostname.txt);
 // per ogni
 nome contenuto nella li
sta che avevo creato
do host $ip.example2.eu
|grep -v "not found";
// esegui il commando “host” (esegue un DNS lookup) del nome presente
nella mia list
a, unito al dominio target (
example2.eu). –v
 scarta le
stringhe di output che contengono “not found”.  done  //fine
```

Il risultato fu che, fra gli n server che trovai, vi era un **webmail.example2.eu** ancora attivo. Navigando sull’indirizzo IP associato, venivo reindirizzato automaticamente ad un vecchio server di posta[8] ancora attivo.

PARTE V: SMTP

Decisi di fare una ricognizione non invasiva su **webmail.example2.eu** utilizzando **nmap**[9]:

```
nmap -sS –p25 192.168.X.
```

- sS: permette di effettuare la scansione delle porte in modo “relativamente” nascosto e poco invasivo, non completando del tutto la connessione TCP;

- p: specifica esplicitamente la porta sul quale effettuare la scansione. Non sapendo chi c’era dall’altra parte, decisi di procedere con prudenza, specificando solo la porta che mi interessava. Nel mio caso il protocollo **SMTP**, **porta 25**, utilizzato come standard per la trasmissione di e-mail;

- 143 192.168.X.X: era l’IP del mio target: **webmail.example2.eu**.

Il risultato del comando diede come esito: “SERVICE: smtp STATE: open”.

A questo punto, con la porta 25 in “state open”, avrei potuto utilizzare diversi script per l’enumerazione degli utenti (ad esempio: smtp-user-enum) con wordlist specifiche, ma avevo scarse informazioni sui potenziali utenti di quel server. Quindi decisi di fare un tentativo con

alcune credenziali di default.

Utilizzai ancora il mio terminale e digitai:

```
telnet 192.168.X.X 25
```

Per chi non lo sapesse, telnet è un protocollo “generalizzato” che viene utilizzato solitamente per aprire sessioni di login[10].

La sessione mi restituì “Connected to **webmail.example2.eu”** e quello che in gergo viene chiamato “Banner” (contenente molte volte diverse informazioni sensibili). Il mio primo comando fu un banale **HELO** che dava inizio alle comunicazioni fra il mio client telnet e il server **SMTP.**

Utilizzai quindi il comando **VRFY** seguito da **administrator** poi **root** e successivamente **sys.** Ma non accadde nulla: le risposte furono tutte negative.

A questo punto, decisi di chiudere la connessione e non tentare ulteriori accessi, per il momento. Avevo bisogno di maggiori informazioni.

PARTE VI: THEHARVESTER

Mi venne in soccorso un programma in grado di estrarre dati da un dato sitoweb e di recuperare informazioni sul target consultando numerose fonti: THEHARVESTER.

Dalla mia Kali, digitai:

```
theharvester –d example2.eu –b all
```

- d specificava il dominio target;

- l’opzione ***–b*** indicava le sorgenti di dati a cui volevo attingere. Ad esempio: yahoo, twitter, linkedin, etc…Nel mio caso “all”, cioè cercai in tutte le sorgenti disponibili.

Il risultato furono una quindicina di indirizzi email. Rimossi quelli delle mailing list[11] (ad esempio amministrazione, paghe, contabilità, etc.), ne rimasero otto. Tutti erano del tipo classico: nome.cognome@example2.com.

La cosa si faceva interessante, ma mi mancavano le password. Avrei potuto tentare un *brute force*[12] classico a questo punto, ma temevo che il server andasse giù o, peggio, che qualcuno si accorgesse dell’attacco. Decisi quindi di seguire ancora la strada dell’OSINT.

Nota di Approfondimento: *un attacco di tipologia brute force porta il server vittima al limite in termini di risorse hardware (anche se esistono diverse tecniche per mitigare l' "overload" generato), il risultato potrebbe essere un crash del sistema. La possibilità che l'evento avvenga è potenzialmente maggiore su macchine datate.*

PARTE VII: DATA BREACH

Avevo diverse possibilità. Fra queste, ad esempio, avrei potuto impiegare tool come Maltego per la ricerca di fingerprint[13]. Decisi invece di verificare se gli indirizzi mail fossero stati esposti ad un Data Breach[14], utilizzando siti come: https://haveibeenpwned.com/.

Quando vidi comparire la scritta *"Oh no — pwned!"* Scaricai le collection[15] contenenti tutti i Breach, e quindi le password delle mail compromesse. Ovviamente nei file non erano presenti password in chiaro, bensì gli Hash[16] delle password stesse. A questo punto era giunto il momento di tentare il *brute force* in locale sulla mia lista di hash, appena scaricata.

Per farlo utilizzai hashcat https://hashcat.net/hashcat/ su Kali.

Vi faccio un esempio:

hashcat -m 0 -a 0 -o cracked.txt target.lst txt/wordlistExample2_eu.txt

- *m:* identifica il format dell'hash da craccare (MD5 in questo caso);

- *a 0:* un attacco dizionario;

- *o:* l'output con le password;

- *target.lst:* lista degli hash;

- la *word list* che avevo preparato per l'occasione.

Il lasso di tempo, avendo anche a disposizione l'hardware dedicato, fu notevole. La mia speranza era questa: che il tassello che mancava fosse la correlazione fra la password e il mio username.

Avevo preparato una word list utilizzando un *crunch*[17] (se siete interessati, vi rimando a un mio precedente articolo: A case history - Worst Case Scenario) con molteplici combinazioni di password contenente i nomi che avevo a disposizione della holding e dei vari ex impiegati. Il cracking sulle password sarebbe andato a buon fine solo se il match - fra la mia word list e le password contenute nella *collection deldata breach* - avesse trovato una correlazione 1:1. Un tentativo al limite, ma decisi comunque di tentare.

Saltarono fuori due password. Una era composta da tre lettere del nome, tutto il cognome con una maiuscola finale e un punto esclamativo. L'associazione del cognome fu scontata. La seconda era un classico: tutto il cognome, con la prima lettera in maiuscolo, e un punto

interrogativo finale. Le associazioni furono pressoché immediate.
Incrociai le dita e tornai sul server Exchange: **webmail.example2.eu**, digitandovi user e password.

Fu così che dalle email successivamente rinvenute e analizzate dagli investigatori venne fuori un caso di vendetta. L'ex amministratore delegato della holding ritenuta responsabile (Example2 S.p.A.) era finito sotto inchiesta, per volere del nostro "personaggio noto". Aveva quindi ingaggiato un gruppo di cyber-criminali al fine di estorcere dati sensibili alla vittima e diffonderli sul web, per diffamarlo pubblicamente. La polizia postale, che eseguì il *forensic* sui dispositivi, confermò che erano stati hackerati e che la rubrica con tutti i numeri della vittima era stata data in pasto alla rete, comprese le foto dei due figli.

Referenze:

1. https://gist.github.com/FiloSottile/3787073
2. https://it.wikipedia.org/wiki/Open_Source_Intelligence
3. https://it.wikipedia.org/wiki/Dark_web
4. https://it.wikipedia.org/wiki/Kali_Linux
5. https://it.wikipedia.org/wiki/Bash
6. https://addons.mozilla.org/it/firefox/addon/resurrect-pages/
7. https://www.html.it/pag/45642/gli-strumenti-f12-il-debug-con-ie11/
8. https://it.wikipedia.org/wiki/Microsoft_Exchange
9. https://it.wikipedia.org/wiki/Nmap
10. https://it.wikipedia.org/wiki/Mailing_list
11. https://it.wikipedia.org/wiki/Telnet
12. https://it.wikipedia.org/wiki/Metodo_forza_bruta
13. https://en.wikipedia.org/wiki/Maltego
14. https://en.wikipedia.org/wiki/Data_breach
15. https://en.wikipedia.org/wiki/Collection_No._1
16. https://it.wikipedia.org/wiki/Funzione_crittografica_di_hash
17. https://www.ictsecuritymagazine.com/articoli/a-case-history-worst-case-scenario/

Articolo a cura di **Vincenzo Digilio**