Diving Deeper Into Pre-created Computer Accounts

optiv.com/insights/source-zero/blog/diving-deeper-pre-created-computer-accounts

February 06, 2023

Establishing a foothold during internal engagements has become increasingly difficult, as organizations are more aware of security threats and have taken measures to protect their systems from unauthorized access. As a pentester, this predicament challenges you to test your critical thinking skills to come up with new approaches. Back in May of 2022, Oddvar Moe published a blog detailing cases where pre-created or staged computer accounts may be configured with a default password and potentially leveraged for initial access. Researching this further revealed these types of accounts are a systemic issue observed in numerous different environments.

This article provides a high-level summary of pre-Windows 2000 machine accounts, a technical deep dive into scenarios where administrators inadvertently configure accounts with a default password, and a demonstration of how these accounts can be used to bypass restrictions to join a rogue host to a domain.

Pre-Windows 2000 Compatible Computers

In an Active Directory (AD) network, Administrators will often stage computer accounts to ensure the host is in the proper organizational unit or security group prior to joining the host(s) to the domain. If the staged computer account is configured as pre-Windows 2000 compatible, they are set with a password that matches the hostname in all lowercase. For example: **HOSTNAME$:hostname**.

Oddvar's blog highlights some specifics on how these accounts could be identified and used:

1. The account will have a user account control (UAC) attribute of 4128
2. The account will have a logoncount attribute value of 0
3. The account can be used for authentication following a password change or via Kerberos without a password change.

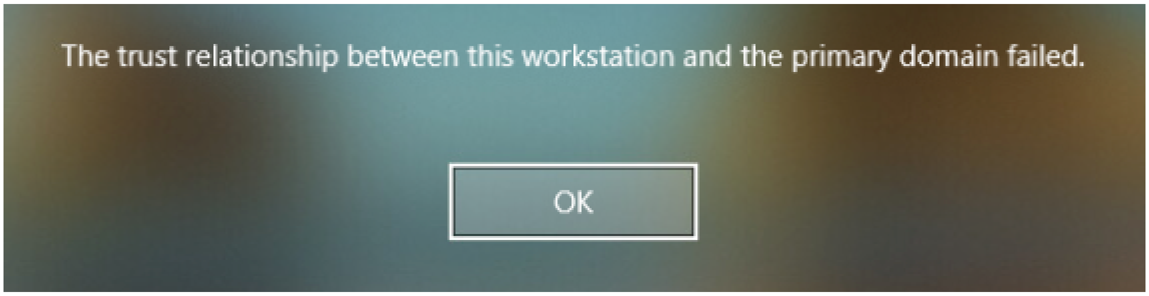
Across multiple engagements, nearly every account discovered matched these details. However, accounts were observed with different UAC values and/or with logoncount values numbering in the thousands. Even more interesting, some of these hosts were still live in the clients' network. This resulted in two questions:

1. Why are these accounts behaving as pre-Windows 2000 compatible machines, but they are live on the network?
2. Why are they valid for Kerberos authentication without a password change?

Insecure Default Configuration

As mentioned previously, several accounts were observed configured with default passwords that were active on the network. This went against common understanding of computer account passwords. Typically, a machine joined to a domain would possess a randomized 120-character password that rotates every 30 days. If the account were rotating correctly, the password should have been different from the original default value. For example, one machine was running a version of Windows Server and had a logon count over 4,000, which indicated it was actively in use. The account was added in April 2015 and likely followed the default 30-day password rotation until the final password change in November 2017. The first scenario that comes to my mind with a gap in password changes this long is that this machine would likely have a broken trust relationship with the domain controller (DC). In some cases, a trust relationship breaks between a host and a DC when the machine resets its password but has lost contact with the DC and the change is not replicated. This creates a conflict between what is stored in AD for the machine's password and what is stored on the host, ultimately breaking authentication. This suspicion was confirmed following attempts to authenticate to the identified hosts that were met with a broken trust relationship error.

Image

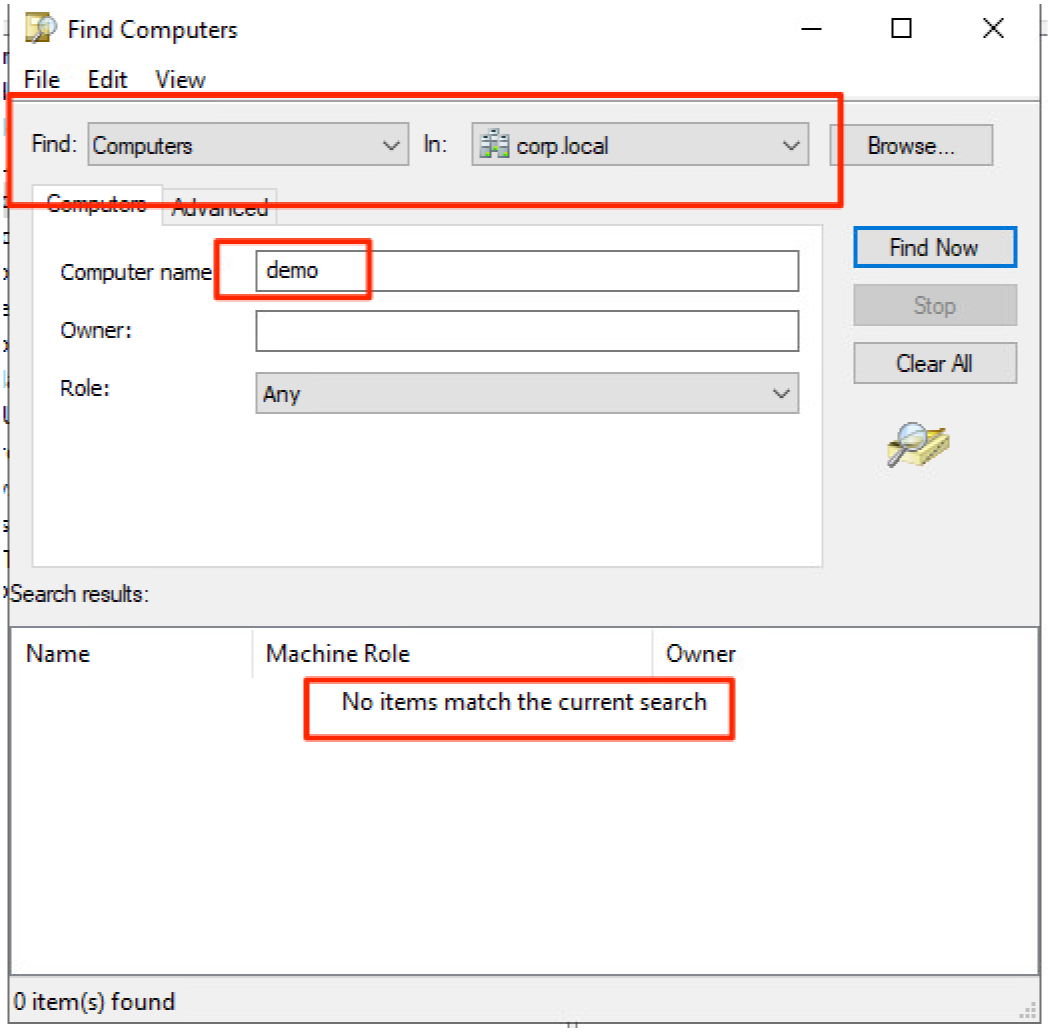


The trust relationship between this workstation and the primary domain failed.

Figure 1: Broken Trust

At this point, researchers tested their theories of what might have happened in a lab environment—specifically, scenarios where hosts would break trusts with the domain. To set the stage, and to highlight the domain join process, the DEMO workstation does not exist in the fictitious corp.local domain.

Image



Find: Computers In: corp.local

Computer name: demo

Owner:

Role: Any

Search results:

Name	Machine Role	Owner

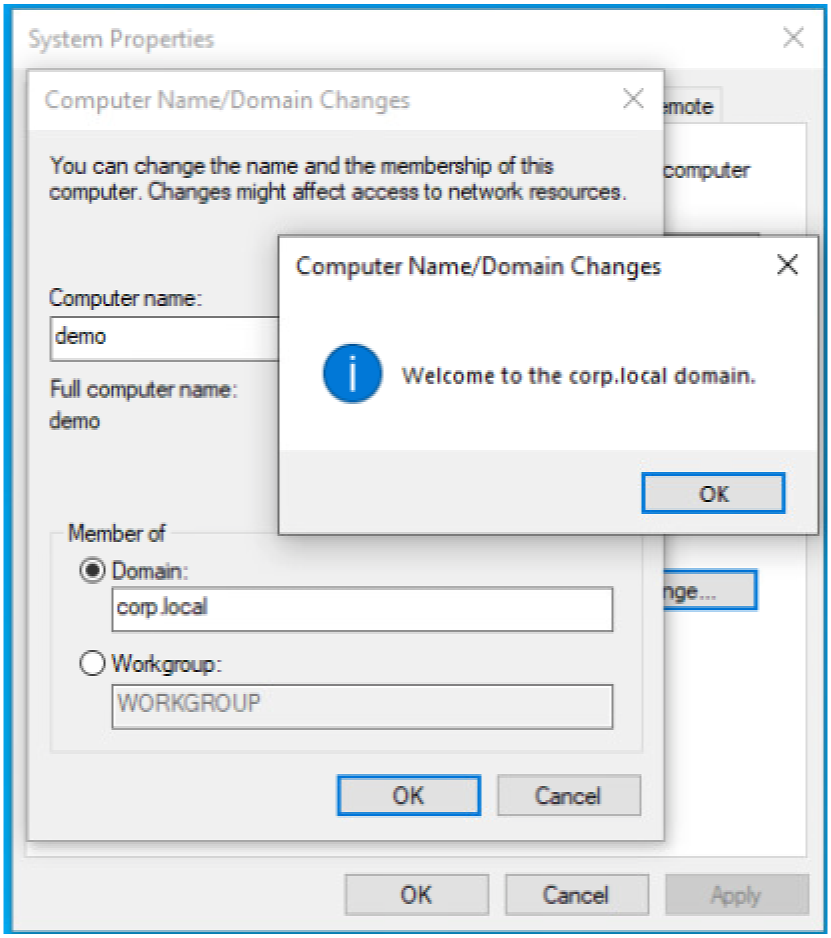
No items match the current search

0 item(s) found

Figure 2: DEMO Account Doesn't Exist

Next, the DEMO account host is joined to the domain via the control panel.

Image



System Properties

Computer Name/Domain Changes

You can change the name and the membership of this computer. Changes might affect access to network resources.

Computer name:
demo

Full computer name:
demo

Member of

● Domain:
corp.local

○ Workgroup:
WORKGROUP

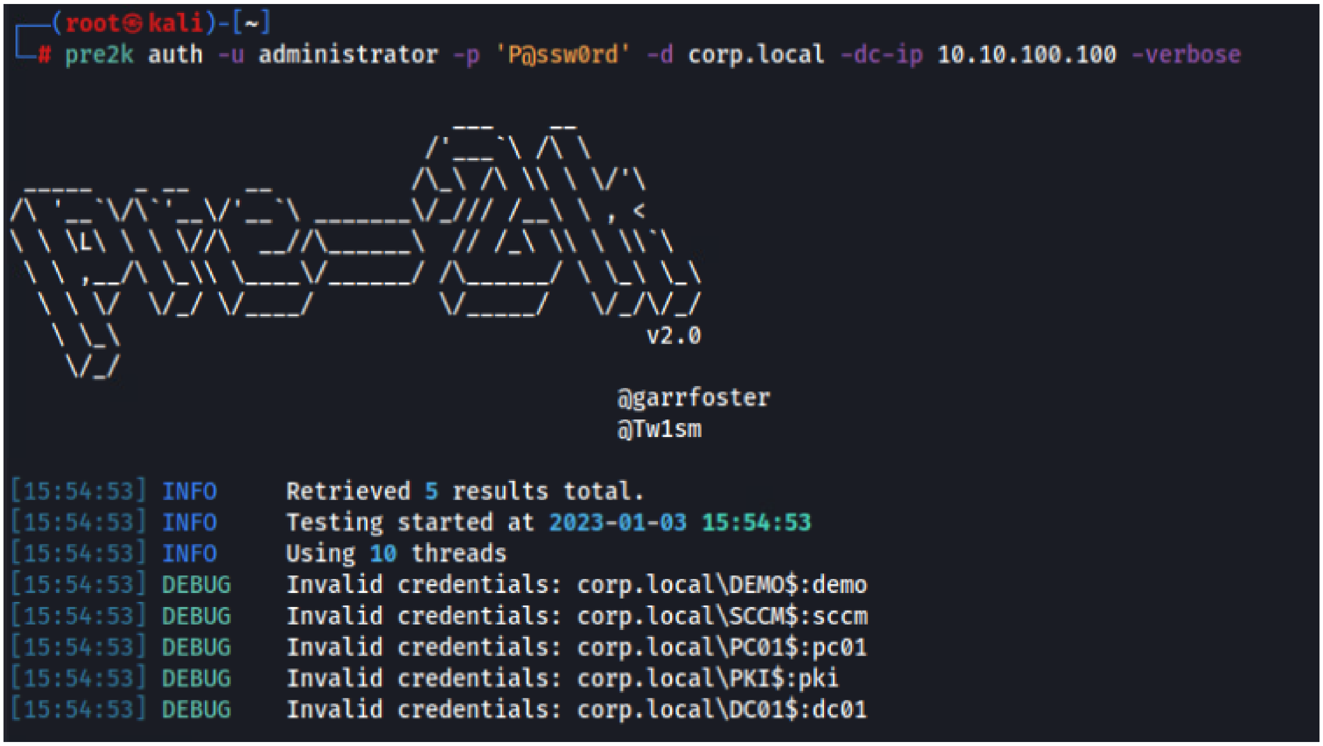
Computer Name/Domain Changes

Welcome to the corp.local domain.

Figure 3: Domain Join

Rather than wait for the host to lose trust, the computer account was reset in Active Directory Users and Computers to simulate the broken trust relationship. Prior to resetting, the Pre2k tool was run for every host in the test environment to set a baseline. Pre2k is a tool I created that can identify pre-Windows 2000 compatible accounts. As seen in the screenshot below, none of the hosts in the domain are configured with a default password.

Image

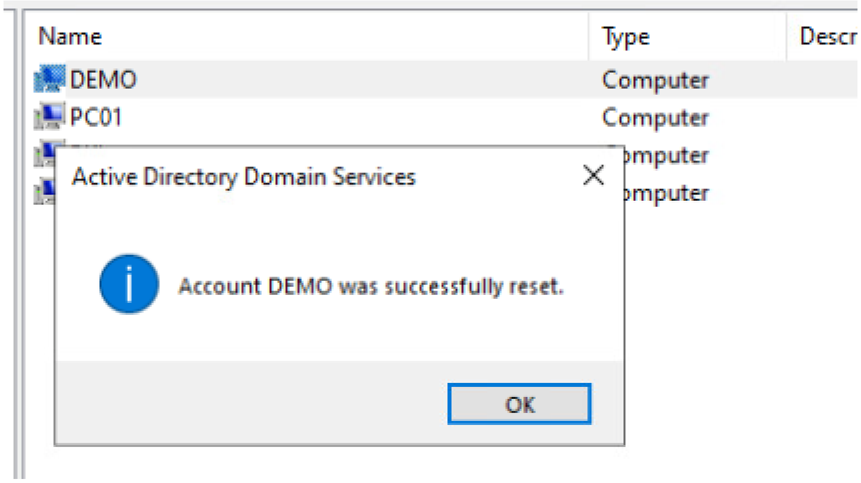


```
┌──(root㉿kali)-[~]
└─# pre2k auth -u administrator -p 'P@ssw0rd' -d corp.local -dc-ip 10.10.100.100 -verbose

                               ___    __
                             /'___`\ /\ \
 _____   _ __    __         /\_\ /\ \\ \ \/'\
/\ '__`\/\`'__\/'__`\ ______\/_/// /__\ \ , <
\ \ \L\ \ \ \//\  __//\______\  // /_\ \\ \ \\`\
 \ \ ,__/\ \_\\ \____\/______/ /\______/ \ \_\ \_\
  \ \ \/  \/_/ \/____/         \/_____/   \/_/\/_/
   \ \_\                                        v2.0
    \/_/

                                              @garrfoster
                                              @Tw1sm

[15:54:53] INFO     Retrieved 5 results total.
[15:54:53] INFO     Testing started at 2023-01-03 15:54:53
[15:54:53] INFO     Using 10 threads
[15:54:53] DEBUG    Invalid credentials: corp.local\DEMO$:demo
[15:54:53] DEBUG    Invalid credentials: corp.local\SCCM$:sccm
[15:54:53] DEBUG    Invalid credentials: corp.local\PC01$:pc01
[15:54:53] DEBUG    Invalid credentials: corp.local\PKI$:pki
[15:54:53] DEBUG    Invalid credentials: corp.local\DC01$:dc01
```

Figure 4: Invalid Credentials

Next, the computer account for the DEMO$ host was reset to break the trust relationship and Pre2k was ran again. Following the reset, the DEMO$ account was now configured with a default password.

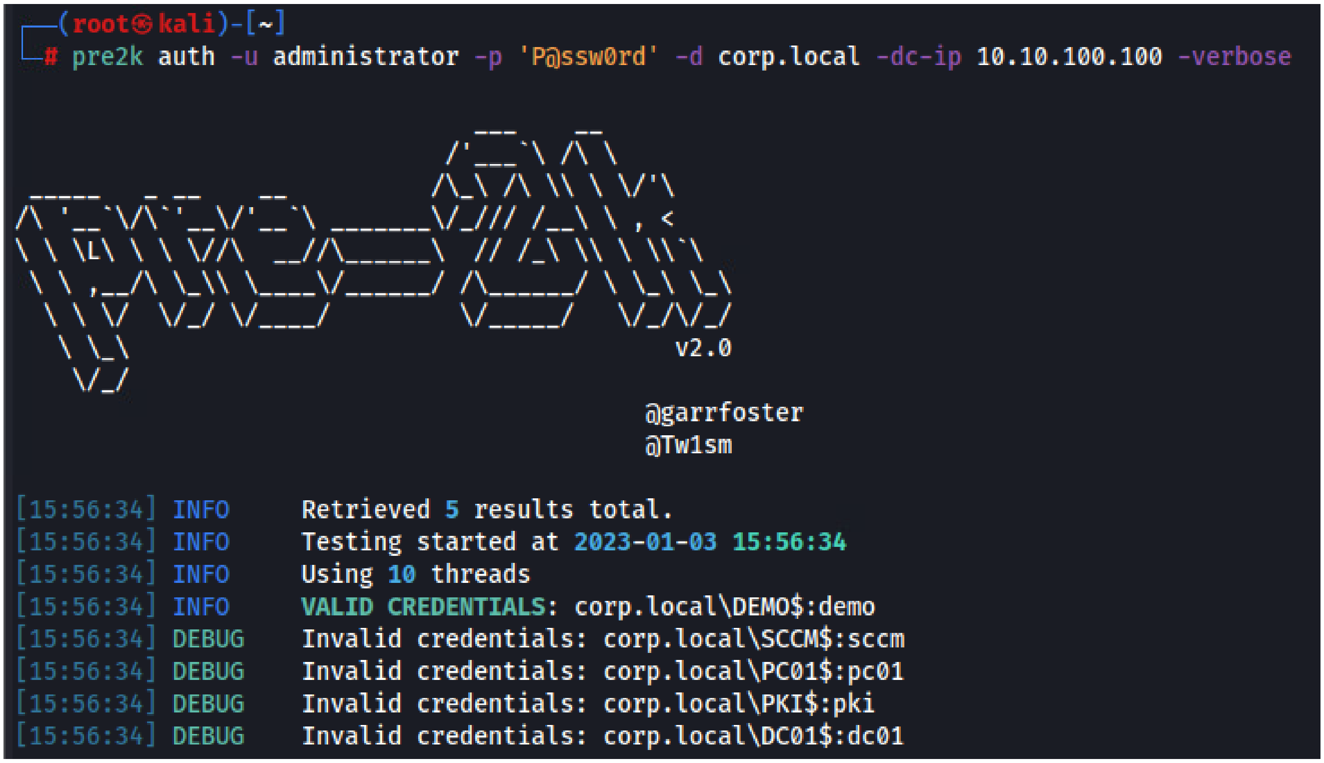
Image



Account DEMO was successfully reset.

Figure 5: Reset Account

Image

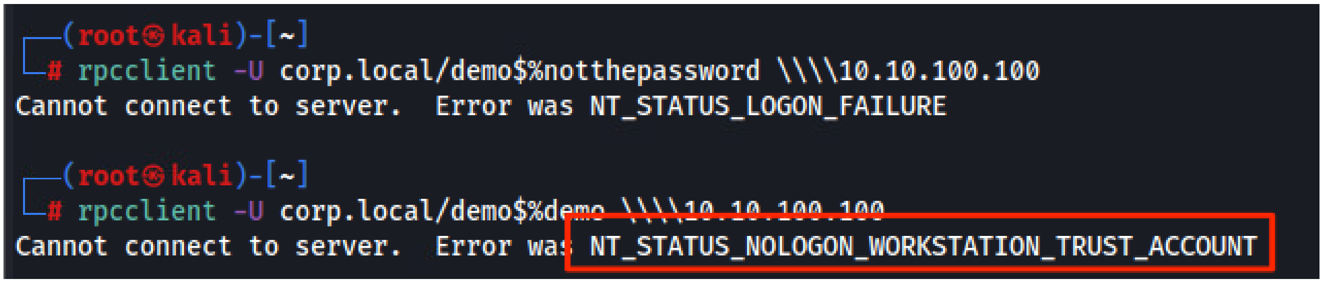


```
┌──(root㉿kali)-[~]
└─# pre2k auth -u administrator -p 'P@ssw0rd' -d corp.local -dc-ip 10.10.100.100 -verbose

                                ___    __
                              /'___`\ /\ \
 _____   _ __    __          /\_\ /\ \\ \ \/'\
/\ '__`\/\`'__\/'__`\ _______\/_/// /__\ \ , <
\ \ \L\ \ \ \//\  __//\______\  // /_\ \\ \ \\`\
 \ \ ,__/\ \_\\ \____\/_______/ /\______/ \ \_\ \_\
  \ \ \/  \/_/ \/____/         \/_____/   \/_/\/_/
   \ \_\                                      v2.0
    \/_/

                                            @garrfoster
                                            @Tw1sm

[15:56:34] INFO     Retrieved 5 results total.
[15:56:34] INFO     Testing started at 2023-01-03 15:56:34
[15:56:34] INFO     Using 10 threads
[15:56:34] INFO     VALID CREDENTIALS: corp.local\DEMO$:demo
[15:56:34] DEBUG    Invalid credentials: corp.local\SCCM$:sccm
[15:56:34] DEBUG    Invalid credentials: corp.local\PC01$:pc01
[15:56:34] DEBUG    Invalid credentials: corp.local\PKI$:pki
[15:56:34] DEBUG    Invalid credentials: corp.local\DC01$:dc01
```

Figure 6: Valid Credential Following Reset

From a Linux operating system, the first command below was used to demonstrate a typical logon failure as the corp\demo$ machine account using an intentionally invalid password. The second command was used to identify that the account was configured with a default password.

Image

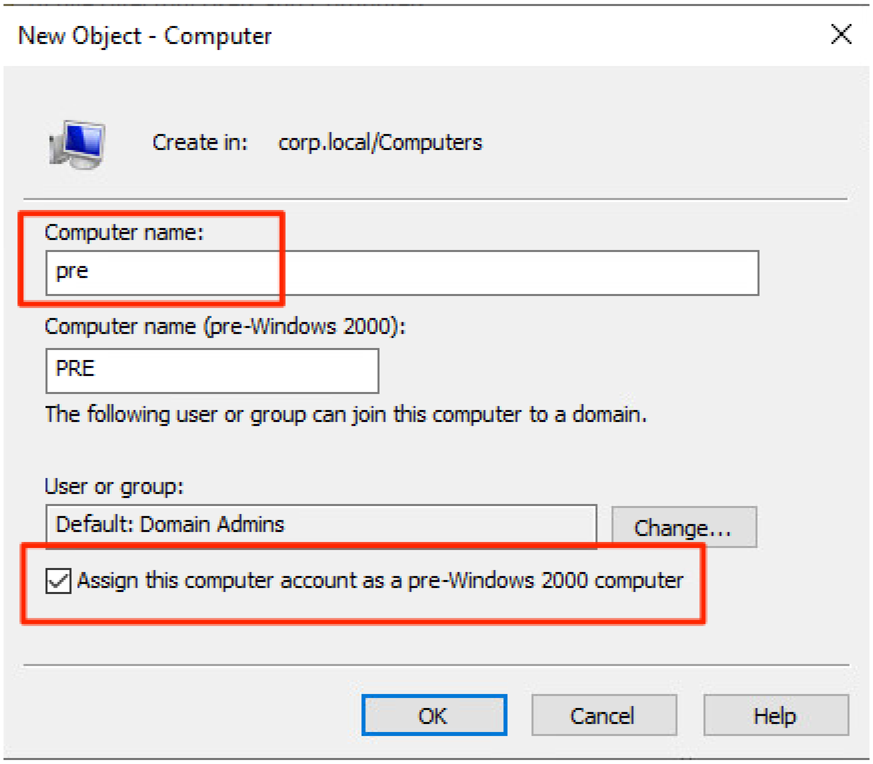


```
┌──(root㉿kali)-[~]
└─# rpcclient -U corp.local/demo$%notthepassword \\\\10.10.100.100
Cannot connect to server.  Error was NT_STATUS_LOGON_FAILURE

┌──(root㉿kali)-[~]
└─# rpcclient -U corp.local/demo$%demo \\\\10.10.100.100
Cannot connect to server.  Error was NT_STATUS_NOLOGON_WORKSTATION_TRUST_ACCOUNT
```

Figure 7: Authentication Errors

The error response for the second login returned a "NT_STATUS_NOLOGON_WORKSTATION_TRUST_ACCOUNT" or a UAC value of 4128. This response indicates the account was configured as pre-Windows 2000 compatible. However, when the DEMO$ account was reset, the UAC value persisted as "WORKSTATION_TRUST_ACCOUNT" or a UAC value of 4096, which conflicted with the previous error response. To demonstrate this conflict, the machine account “PRE$” was created and configured as pre-Windows 2000 compatible and then compared to the DEMO$ account following the reset.

Image



New Object - Computer

Create in: corp.local/Computers

Computer name: pre

Computer name (pre-Windows 2000): PRE

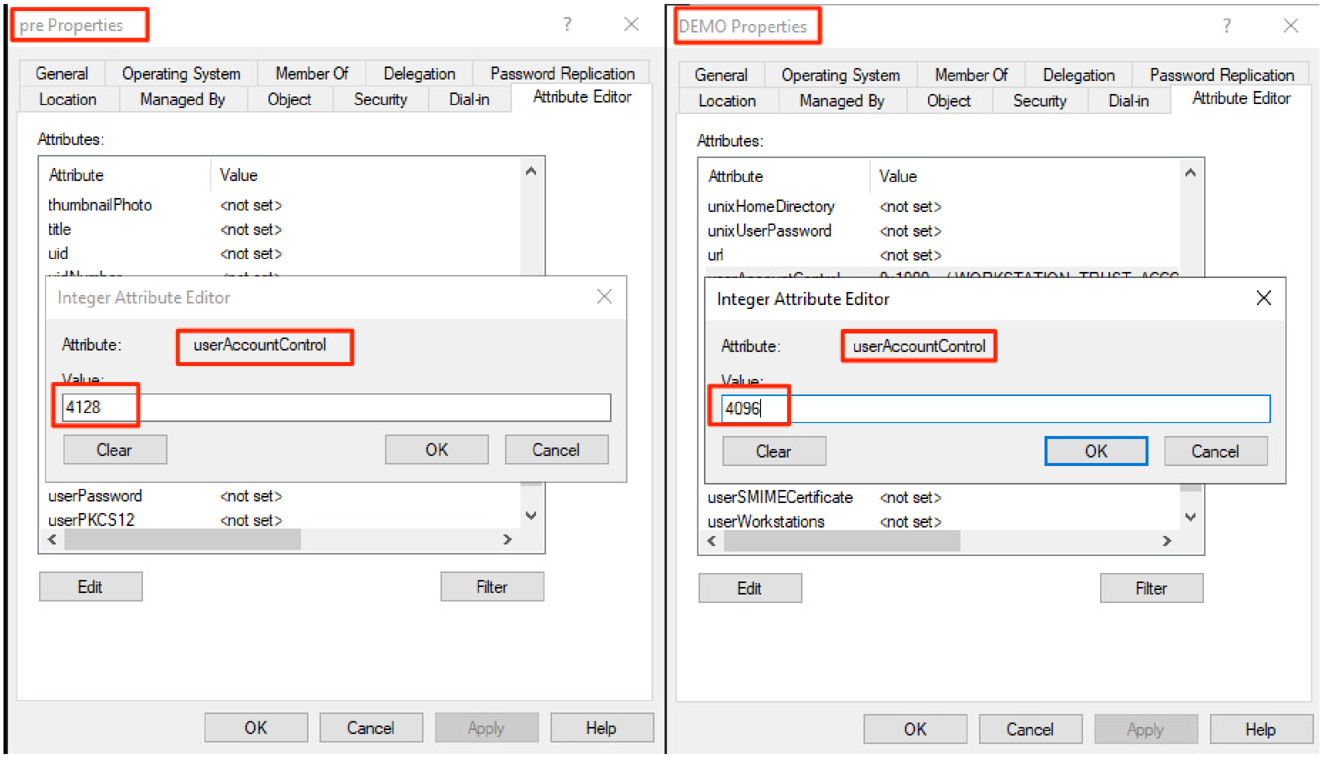
The following user or group can join this computer to a domain.

User or group: Default: Domain Admins

☑ Assign this computer account as a pre-Windows 2000 computer

Figure 8: Pre-created Computer Account

Image



pre Properties

Attribute	Value
thumbnailPhoto	<not set>
title	<not set>
uid	<not set>
userPassword	<not set>
userPKCS12	<not set>

Attribute: userAccountControl

Value: 4128

DEMO Properties

Attribute	Value
unixHomeDirectory	<not set>
unixUserPassword	<not set>
url	<not set>
userSMIMECertificate	<not set>
userWorkstations	<not set>

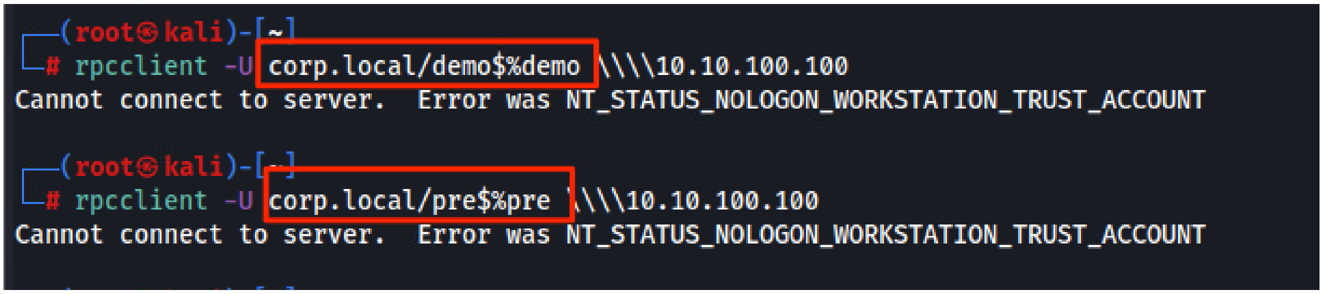
Attribute: userAccountControl

Value: 4096

Figure 9: Compared UAC Attributes

Additionally, the above rpcclient command was executed for the PRE$ account and returned the same error response as the reset DEMO$ account, confirming the reset command reverts to pre-Windows 2000 compatible.

Image



```
┌──(root㉿kali)-[~]
└─# rpcclient -U corp.local/demo$%demo \\\\10.10.100.100
Cannot connect to server.  Error was NT_STATUS_NOLOGON_WORKSTATION_TRUST_ACCOUNT

┌──(root㉿kali)-[~]
└─# rpcclient -U corp.local/pre$%pre \\\\10.10.100.100
Cannot connect to server.  Error was NT_STATUS_NOLOGON_WORKSTATION_TRUST_ACCOUNT
```

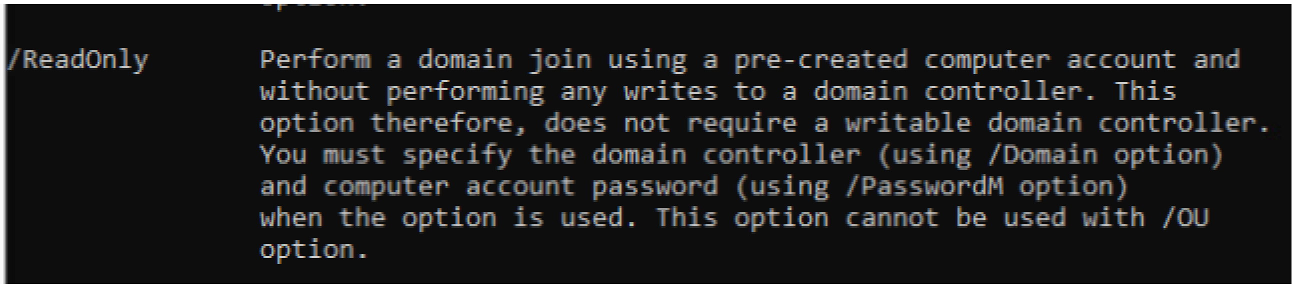
Figure 10: Duplicate Error Response

From here the domain join process was repeated using multiple methods—including PowerShell modules and command line tools—and each had the same result. By resetting the machine account, the account is configured with a default password, regardless of how the account was originally created. Since the UAC value persists through the reset, there is no identifier to associate the reset account to be configured as pre-Windows 2000 compatible. Nor is there any warning to the administrator performing this action that indicates the password is being set to a default value.

Broken Access Control

Mentioned previously, the domain join process was performed using command line tools—one of which was Netdom. Using the Netdom help menu for the join subcommand returned two interesting flags: “ReadOnly” and “PasswordM”. These flags indicated it was possible to perform a domain join using known machine account credentials.s

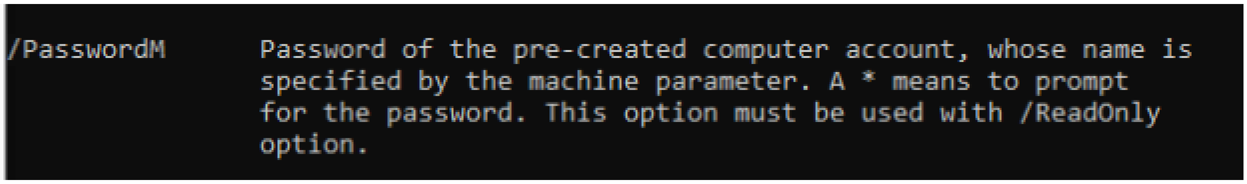
Image



```
/ReadOnly        Perform a domain join using a pre-created computer account and
                 without performing any writes to a domain controller. This
                 option therefore, does not require a writable domain controller.
                 You must specify the domain controller (using /Domain option)
                 and computer account password (using /PasswordM option)
                 when the option is used. This option cannot be used with /OU
                 option.
```

Figure 11: Netdom ReadOnly Flag

Image

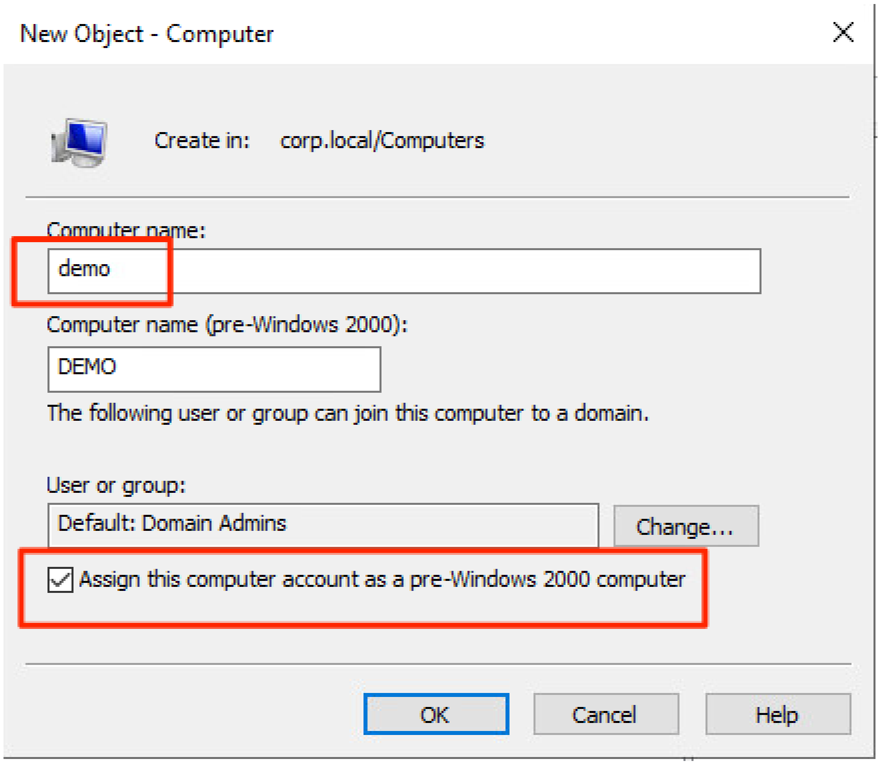


```
/PasswordM       Password of the pre-created computer account, whose name is
                 specified by the machine parameter. A * means to prompt
                 for the password. This option must be used with /ReadOnly
                 option.
```

Figure 12: Netdom PasswordM Flag

To test this functionality, the DEMO account was created with pre-Windows 2000 compatibility and then Netdom was used following the help descriptions to perform the “ReadOnly” domain join, and the process was successful.

Image



New Object - Computer

Create in: corp.local/Computers

Computer name: demo

Computer name (pre-Windows 2000): DEMO

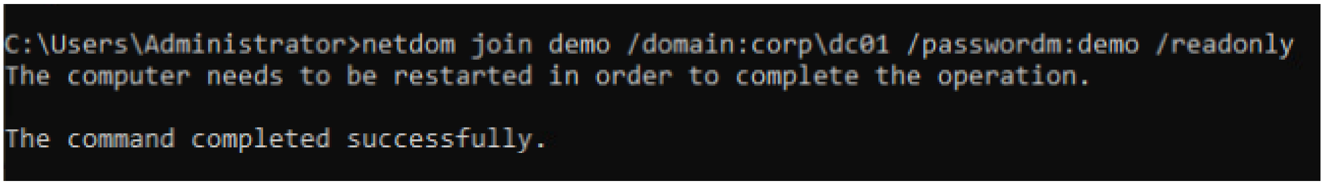
The following user or group can join this computer to a domain.

User or group: Default: Domain Admins

☑ Assign this computer account as a pre-Windows 2000 computer

Figure 13: Pre-created Demo Account

Image

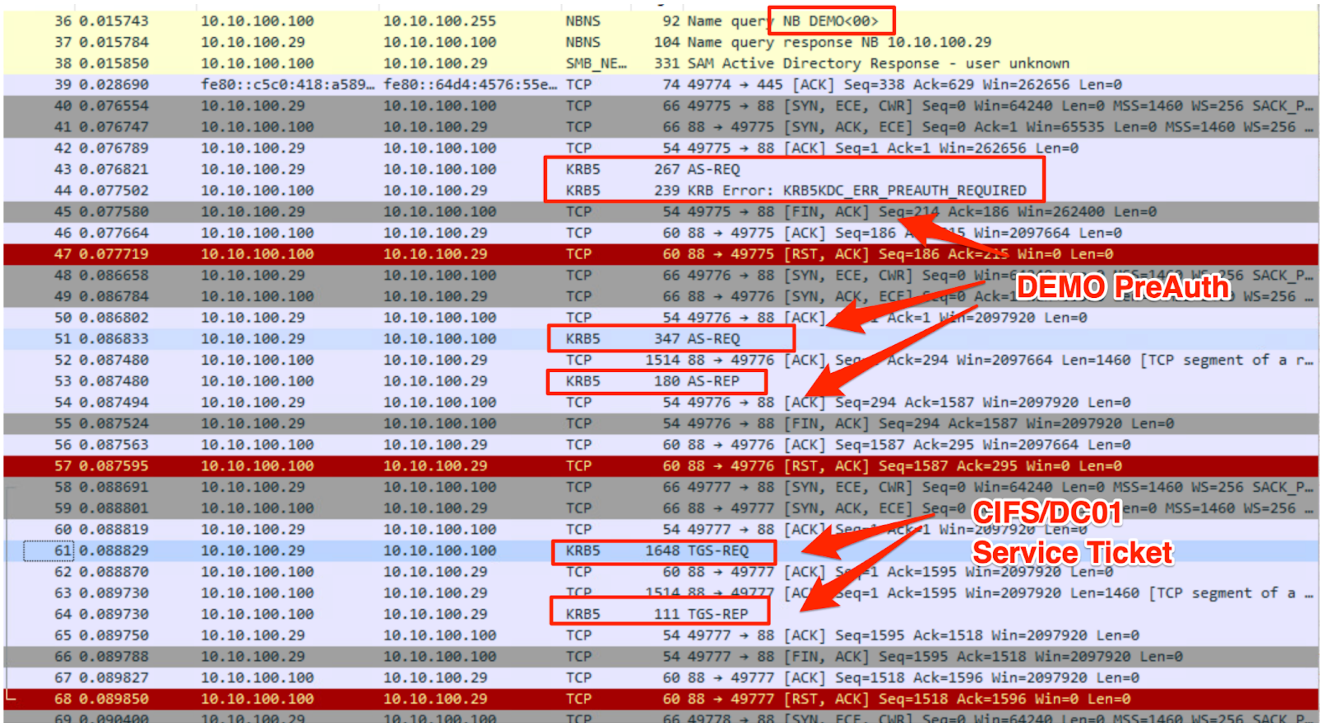


```
C:\Users\Administrator>netdom join demo /domain:corp\dc01 /passwordm:demo /readonly
The computer needs to be restarted in order to complete the operation.

The command completed successfully.
```

Figure 14: Netdom ReadOnly Domain Join

The process was repeated while running a packet capture, which revealed the domain join process with Netdom utilizes Kerberos for authentication.

Image



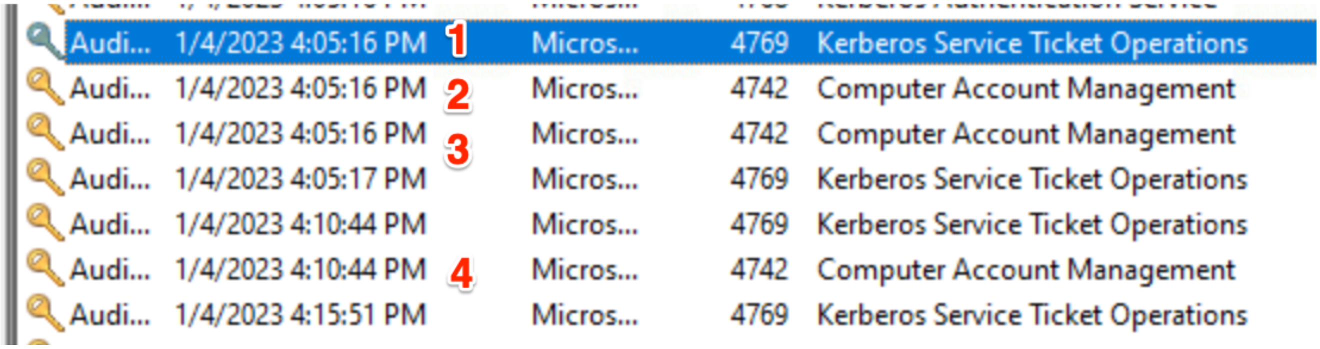
```
36 0.015743  10.10.100.100         10.10.100.255         NBNS      92 Name query NB DEMO<00>
37 0.015784  10.10.100.29          10.10.100.100         NBNS     104 Name query response NB 10.10.100.29
38 0.015850  10.10.100.100         10.10.100.29          SMB_NE…  331 SAM Active Directory Response - user unknown
39 0.028690  fe80::c5c0:418:a589…  fe80::64d4:4576:55e…  TCP       74 49774 → 445 [ACK] Seq=338 Ack=629 Win=262656 Len=0
40 0.076554  10.10.100.29          10.10.100.100         TCP       66 49775 → 88 [SYN, ECE, CWR] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_P…
41 0.076747  10.10.100.100         10.10.100.29          TCP       66 88 → 49775 [SYN, ACK, ECE] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 …
42 0.076789  10.10.100.29          10.10.100.100         TCP       54 49775 → 88 [ACK] Seq=1 Ack=1 Win=262656 Len=0
43 0.076821  10.10.100.29          10.10.100.100         KRB5     267 AS-REQ
44 0.077502  10.10.100.100         10.10.100.29          KRB5     239 KRB Error: KRB5KDC_ERR_PREAUTH_REQUIRED
45 0.077580  10.10.100.29          10.10.100.100         TCP       54 49775 → 88 [FIN, ACK] Seq=214 Ack=186 Win=262400 Len=0
46 0.077664  10.10.100.100         10.10.100.29          TCP       60 88 → 49775 [ACK] Seq=186 Ack=215 Win=2097664 Len=0
47 0.077719  10.10.100.100         10.10.100.29          TCP       60 88 → 49775 [RST, ACK] Seq=186 Ack=215 Win=0 Len=0
48 0.086658  10.10.100.29          10.10.100.100         TCP       66 49776 → 88 [SYN, ECE, CWR] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_P…
49 0.086784  10.10.100.100         10.10.100.29          TCP       66 88 → 49776 [SYN, ACK, ECE] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 …
50 0.086802  10.10.100.29          10.10.100.100         TCP       54 49776 → 88 [ACK] Seq=1 Ack=1 Win=2097920 Len=0
51 0.086833  10.10.100.29          10.10.100.100         KRB5     347 AS-REQ
52 0.087480  10.10.100.100         10.10.100.29          TCP     1514 88 → 49776 [ACK] Seq=1 Ack=294 Win=2097664 Len=1460 [TCP segment of a r…
53 0.087480  10.10.100.100         10.10.100.29          KRB5     180 AS-REP
54 0.087494  10.10.100.29          10.10.100.100         TCP       54 49776 → 88 [ACK] Seq=294 Ack=1587 Win=2097920 Len=0
55 0.087524  10.10.100.29          10.10.100.100         TCP       54 49776 → 88 [FIN, ACK] Seq=294 Ack=1587 Win=2097920 Len=0
56 0.087563  10.10.100.100         10.10.100.29          TCP       60 88 → 49776 [ACK] Seq=1587 Ack=295 Win=2097664 Len=0
57 0.087595  10.10.100.100         10.10.100.29          TCP       60 88 → 49776 [RST, ACK] Seq=1587 Ack=295 Win=0 Len=0
58 0.088691  10.10.100.29          10.10.100.100         TCP       66 49777 → 88 [SYN, ECE, CWR] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_P…
59 0.088801  10.10.100.100         10.10.100.29          TCP       66 88 → 49777 [SYN, ACK, ECE] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 …
60 0.088819  10.10.100.29          10.10.100.100         TCP       54 49777 → 88 [ACK] Seq=1 Ack=1 Win=2097920 Len=0
61 0.088829  10.10.100.29          10.10.100.100         KRB5    1648 TGS-REQ
62 0.088870  10.10.100.100         10.10.100.29          TCP       60 88 → 49777 [ACK] Seq=1 Ack=1595 Win=2097920 Len=0
63 0.089730  10.10.100.100         10.10.100.29          TCP     1514 88 → 49777 [ACK] Seq=1 Ack=1595 Win=2097920 Len=1460 [TCP segment of a …
64 0.089730  10.10.100.100         10.10.100.29          KRB5     111 TGS-REP
65 0.089750  10.10.100.29          10.10.100.100         TCP       54 49777 → 88 [ACK] Seq=1595 Ack=1518 Win=2097920 Len=0
66 0.089788  10.10.100.29          10.10.100.100         TCP       54 49777 → 88 [FIN, ACK] Seq=1595 Ack=1518 Win=2097920 Len=0
67 0.089827  10.10.100.100         10.10.100.29          TCP       60 88 → 49777 [ACK] Seq=1518 Ack=1596 Win=2097920 Len=0
68 0.089850  10.10.100.100         10.10.100.29          TCP       60 88 → 49777 [RST, ACK] Seq=1518 Ack=1596 Win=0 Len=0
69 0.090400  10.10.100.29          10.10.100.100         TCP       66 49778 → 88 [SYN, ECE, CWR] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_P…
```

Figure 15: Netdom Join Packet Capture

Additionally, the event logs on the DC during this process reveal there were few changes when performing a self-service join.

1. Kerberos authentication
2. Computer password reset
3. Computer service principal names added
4. Computer UAC updated from 4128 to 4096

Image

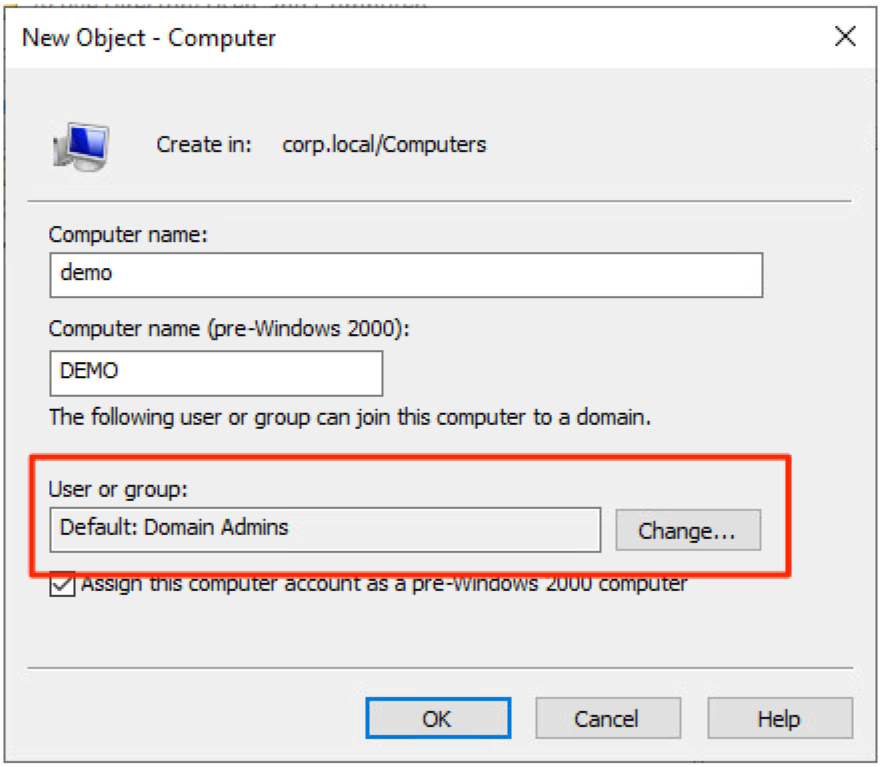


Audi...	1/4/2023 4:05:16 PM	1	Micros...	4769	Kerberos Service Ticket Operations
Audi...	1/4/2023 4:05:16 PM	2	Micros...	4742	Computer Account Management
Audi...	1/4/2023 4:05:16 PM	3	Micros...	4742	Computer Account Management
Audi...	1/4/2023 4:05:17 PM		Micros...	4769	Kerberos Service Ticket Operations
Audi...	1/4/2023 4:10:44 PM		Micros...	4769	Kerberos Service Ticket Operations
Audi...	1/4/2023 4:10:44 PM	4	Micros...	4742	Computer Account Management
Audi...	1/4/2023 4:15:51 PM		Micros...	4769	Kerberos Service Ticket Operations

Figure 16: Event Logs During Domain Join

From these events, it appears this is expected and normal activity. However, something stood out that did not make sense. When pre-creating the machine account, there is an option to specify which principal(s) have the right to join the machine to the domain. By default, this is set to the Domain Administrators group. Yet the machine account was not a member of that group and was able to perform a self-service join. So, it appeared that this ACL was not applying correctly. Furthermore, the group policy was configured to allow only members of the Domain Administrators group to join workstations to the domain. The ms-DS-MachineAccountQuota attribute was set to 0, which should prevent this from occurring.

Image



New Object - Computer

Create in: corp.local/Computers

Computer name:
demo

Computer name (pre-Windows 2000):
DEMO

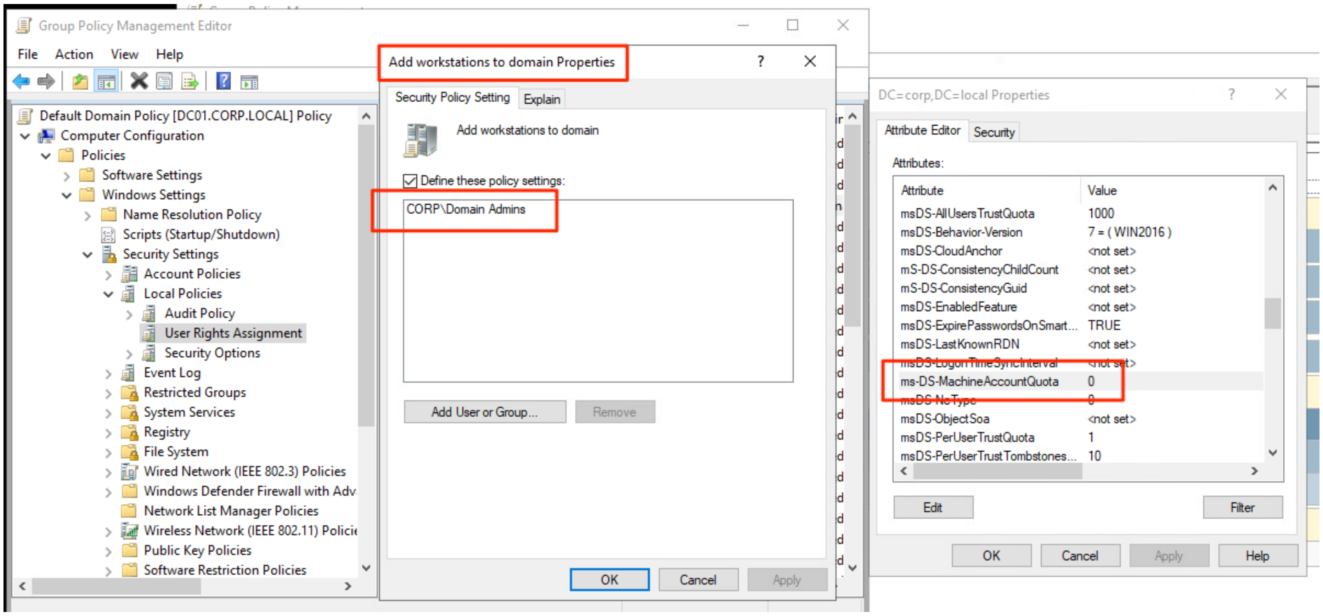
The following user or group can join this computer to a domain.

User or group:
Default: Domain Admins

☑ Assign this computer account as a pre-Windows 2000 computer

Figure 17: Default ACL

Image



Default Domain Policy [DC01.CORP.LOCAL] Policy
- Computer Configuration
  - Policies
    - Software Settings
    - Windows Settings
      - Name Resolution Policy
      - Scripts (Startup/Shutdown)
      - Security Settings
        - Account Policies
        - Local Policies
          - Audit Policy
          - User Rights Assignment
          - Security Options
        - Event Log
        - Restricted Groups
        - System Services
        - Registry
        - File System
        - Wired Network (IEEE 802.3) Policies
        - Windows Defender Firewall with Adv
        - Network List Manager Policies
        - Wireless Network (IEEE 802.11) Policie
        - Public Key Policies
        - Software Restriction Policies

Add workstations to domain Properties

Add workstations to domain

☑ Define these policy settings:

CORP\Domain Admins

DC=corp,DC=local Properties

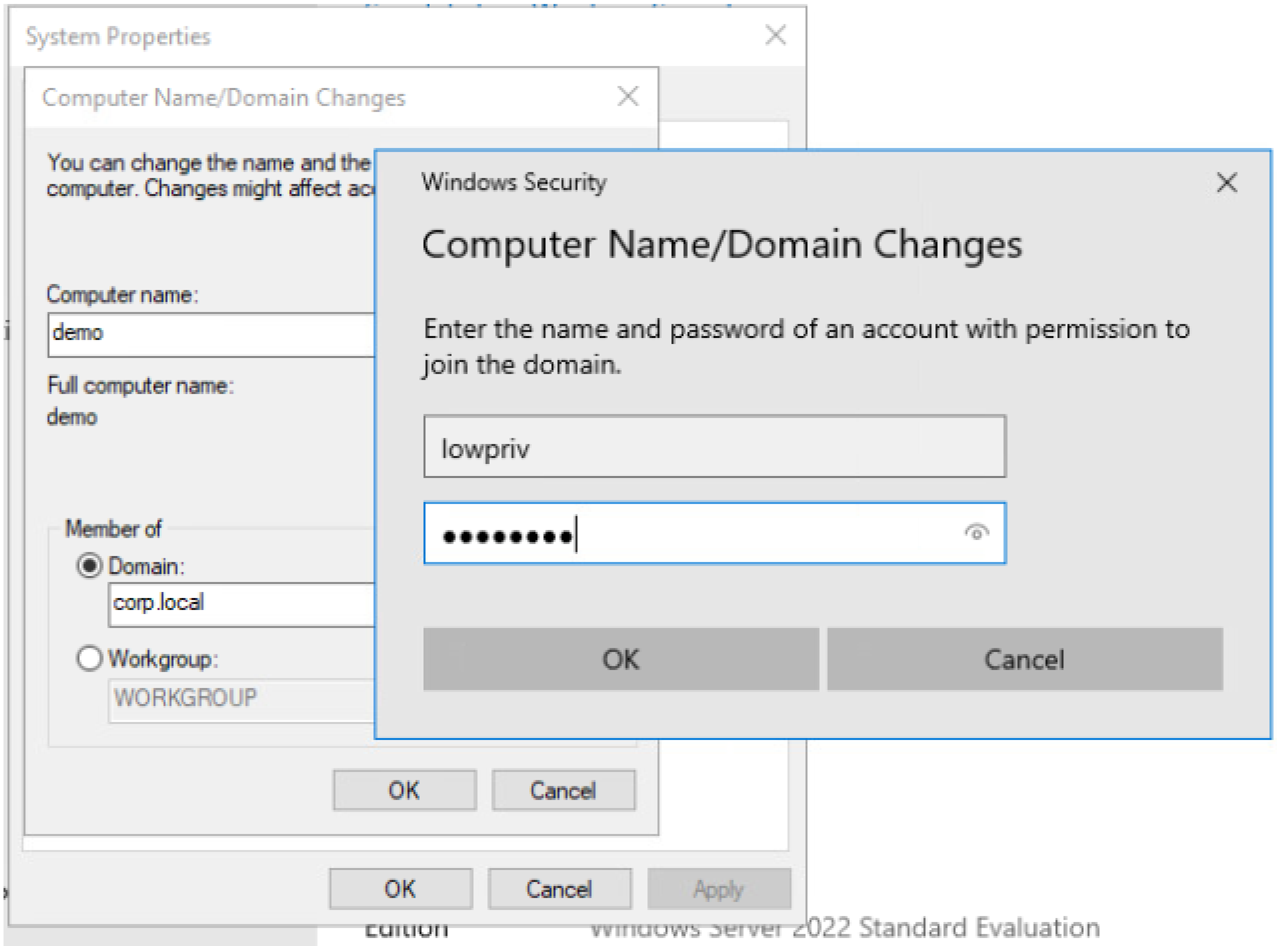
Attributes:

Attribute	Value
msDS-AllUsersTrustQuota	1000
msDS-Behavior-Version	7 = (WIN2016)
msDS-CloudAnchor	<not set>
mS-DS-ConsistencyChildCount	<not set>
mS-DS-ConsistencyGuid	<not set>
msDS-EnabledFeature	<not set>
msDS-ExpirePasswordsOnSmart...	TRUE
msDS-LastKnownRDN	<not set>
msDS-LogonTimeSyncInterval	<not set>
ms-DS-MachineAccountQuota	0
msDS-NcType	0
msDS-ObjectSoa	<not set>
msDS-PerUserTrustQuota	1
msDS-PerUserTrustTombstones...	10

Figure 18: Security Controls

To determine if the controls were applying at all, a domain join was attempted for the DEMO workstation using a low privileged user account through the control panel. That resulted in an "Access Denied" error due to the account being previously created with a different set of credentials. Additionally, a domain join was attempted using the machine account credentials that received the same error.

Image



System Properties

Computer Name/Domain Changes

You can change the name and the computer. Changes might affect ac

Computer name: demo

Full computer name:
demo

Member of

● Domain: corp.local

○ Workgroup: WORKGROUP

Windows Security

Computer Name/Domain Changes

Enter the name and password of an account with permission to join the domain.

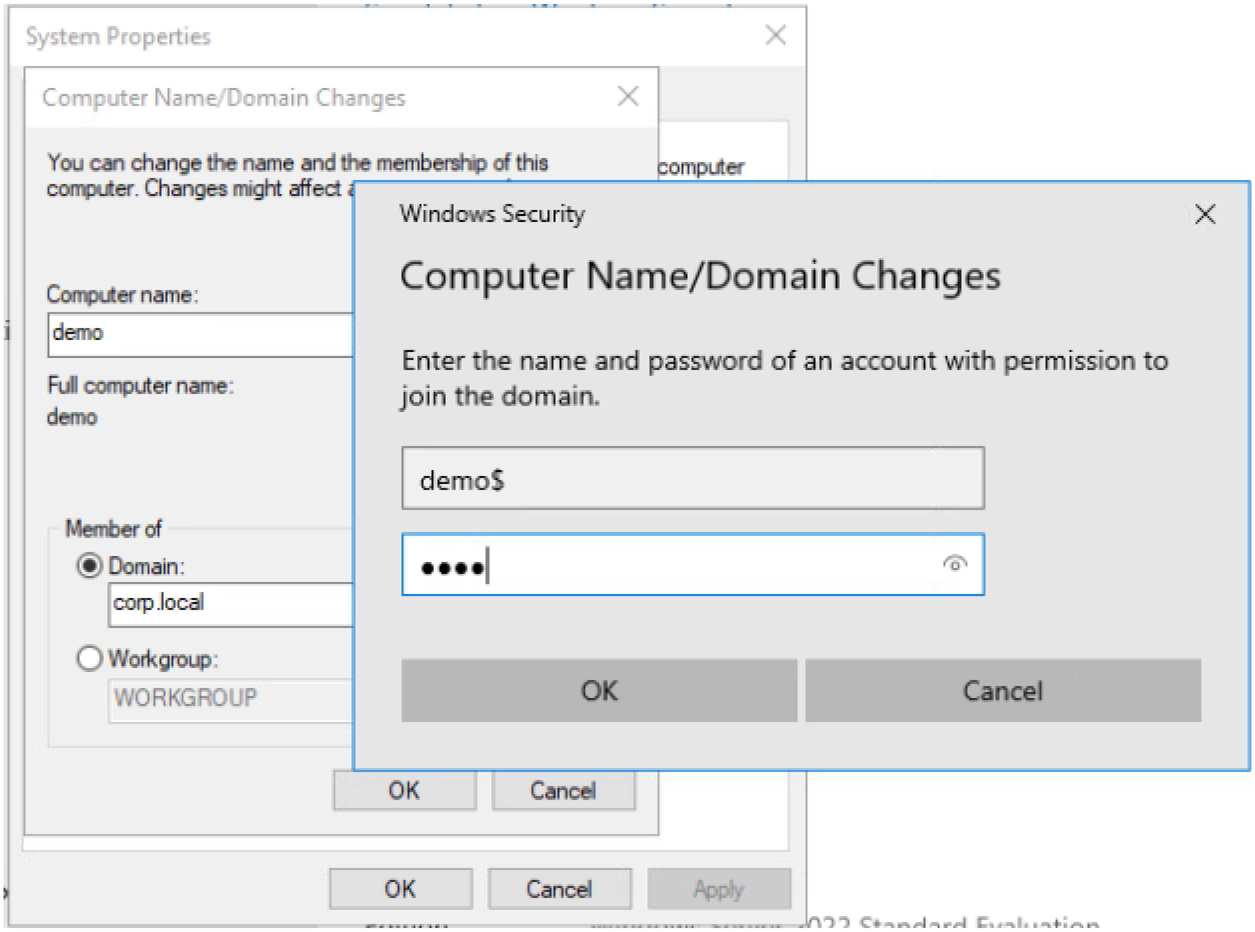
lowpriv

••••••••

Edition Windows Server 2022 Standard Evaluation

Figure 19: Attempted Domain Join with Low Privileged User

Image



You can change the name and the membership of this computer. Changes might affect a

Computer name: demo

Full computer name: demo

● Domain: corp.local

○ Workgroup: WORKGROUP

Computer Name/Domain Changes

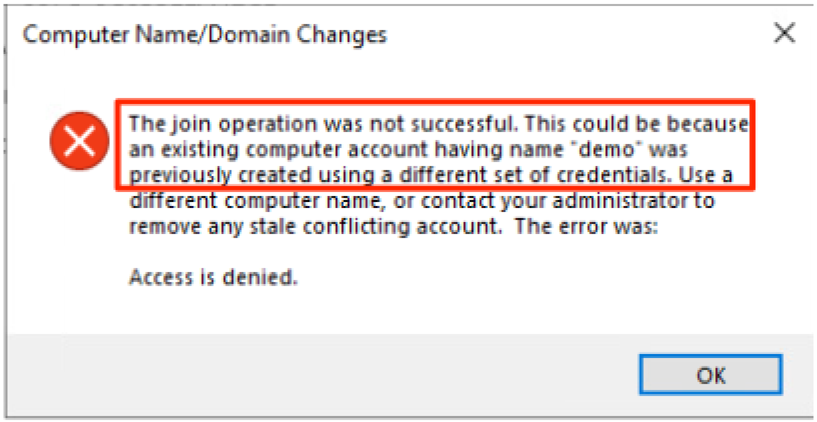
Enter the name and password of an account with permission to join the domain.

demo$

●●●●

Figure 20: Attempted Domain Join with Machine Account

Image



The join operation was not successful. This could be because an existing computer account having name "demo" was previously created using a different set of credentials. Use a different computer name, or contact your administrator to remove any stale conflicting account. The error was:

Access is denied.

Figure 21: Access Denied

Based on these results, a user with knowledge of an existing pre-Windows 2000 machine account can bypass deployed and expected security controls to arbitrarily join a Windows operating system to a domain by performing a “ReadOnly” domain join with Netdom. This action triggers common events that can make detection a challenge.

Conclusion

By resetting a computer account in Active Directory, administrators are unknowingly configuring the accounts with a default password. Upon disclosure to Microsoft, this issue was assigned a moderate severity rating and was scheduled to be fixed at a later date. Moreover, accounts configured in this manner can be used to bypass security controls in a hardened environment that limit the ability to perform domain join operations. In response to this disclosure, Microsoft reviewed it and closed the case.

Mitigations

At the time of writing, I am unaware of an elegant way to identify these accounts. Due to the variances in attribute values, defenders can use Pre2k to perform an authenticated scan of their environment. If accounts are discovered, then the accounts can be disabled if not in use or deleted if not needed. My conclusion is that the only identifying trait for these accounts is the password itself.

In terms of prevention, the best mitigation against exploitation of Pre-Windows 2000 authentication in general is typically considered to be the removal of Anonymous, Everyone, and Authenticated Users from the Pre-Windows 2000 Compatible Access group. That group continues to be included in deployments of new Active Directory forests, even on Windows Server 2022. Removal of Authenticated Users from the group in an existing domain may be a potential challenge in some organizations and should be thoroughly tested. Preferred practice in this context would be to only grant Pre-Windows 2000 compatibility explicitly to specific hosts or users, rather than all authenticated users.

Acknowledgments

These issues would not have been discovered without the collaboration and support of the Attack & Pen team.

Disclosure Timeline

Both issues were reported to MSRC with the following timeline:

1. October 27, 2022 – Reported to MSRC
2. October 28, 2022 – Acknowledged by MSRC
3. November 30, 2022 – Disclosure #1: Vulnerability report assigned moderate severity
4. December 29, 2022 – Disclosure #2: Vulnerability report was closed by MSRC

By:
Garrett Foster