Join me offline!

“Offline Domain Join” in Windows and Samba

sambaXP 2021

Günter Deschner
<gd@samba.org>

Agenda

Offline Domain Join in Windows

- History
- Mechanism and Tools

Offline Domain Join in Samba

- libnetjoin interface
- libnetapi library
- Proposed tooling (net,djoin)

Future tasks:

- Certificate deployment
- GPO deployment

Offline Domain Join

Mechanism and tools in Windows

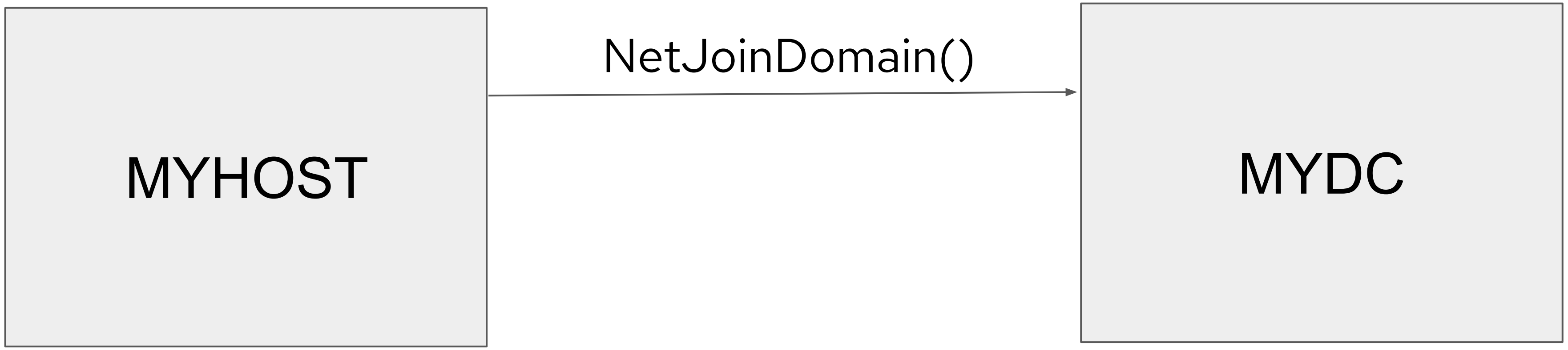
“Classic” Domain Join

Definition and workflow

- What is joining?
- Establish trust relationship between a computer and a domain controller:
  - Create machine account in domain controller database
  - (Optional additional information stored on DC)
  - Set machine account password (shared between computer and DC)
  - Verify successful setup of secure channel between computer and DC

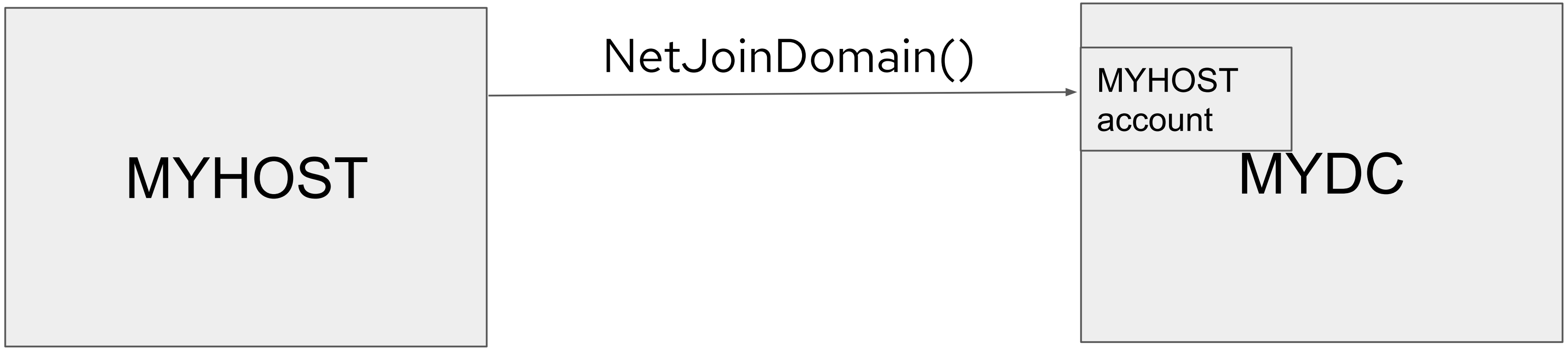
“Classic” Domain Join

Process



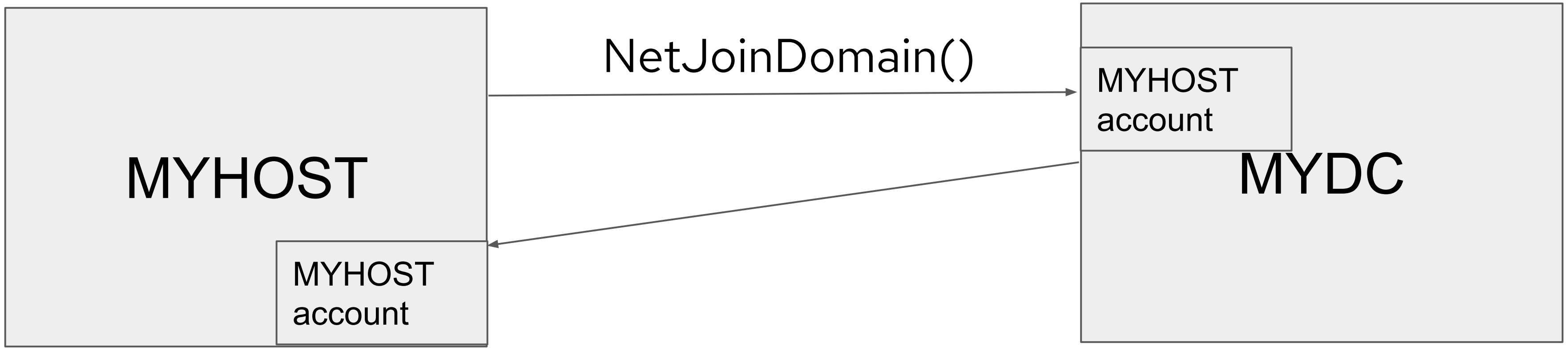
“Classic” Domain Join

Process



“Classic” Domain Join

Process



“Classic” Domain Join

Restrictions

- Computer can only join itself
- Computer needs a running OS for joining
- Network connectivity to DC required and admin credentials
- Difficult to automate or to deploy large numbers of computers
- Later enhancements:
  - Remote Join APIs, using e.g. DCE/RPC wkssvc_NetrJoinDomain2
  - Support for Read-only DCs (complex to setup)

Offline Domain Join (ODJ)

Description

- Available since Windows 7 and Windows 2008 R2 with djoin.exe
- Client and joining computer can be completely separate
- Provisioning and joining are fully separated steps
- DC connectivity is only required for provisioning, not for joining
- Allows joining in locations detached from corporate network
- Provides mass deployment of virtual machines that are then joined during startup

Offline Domain Join

Process

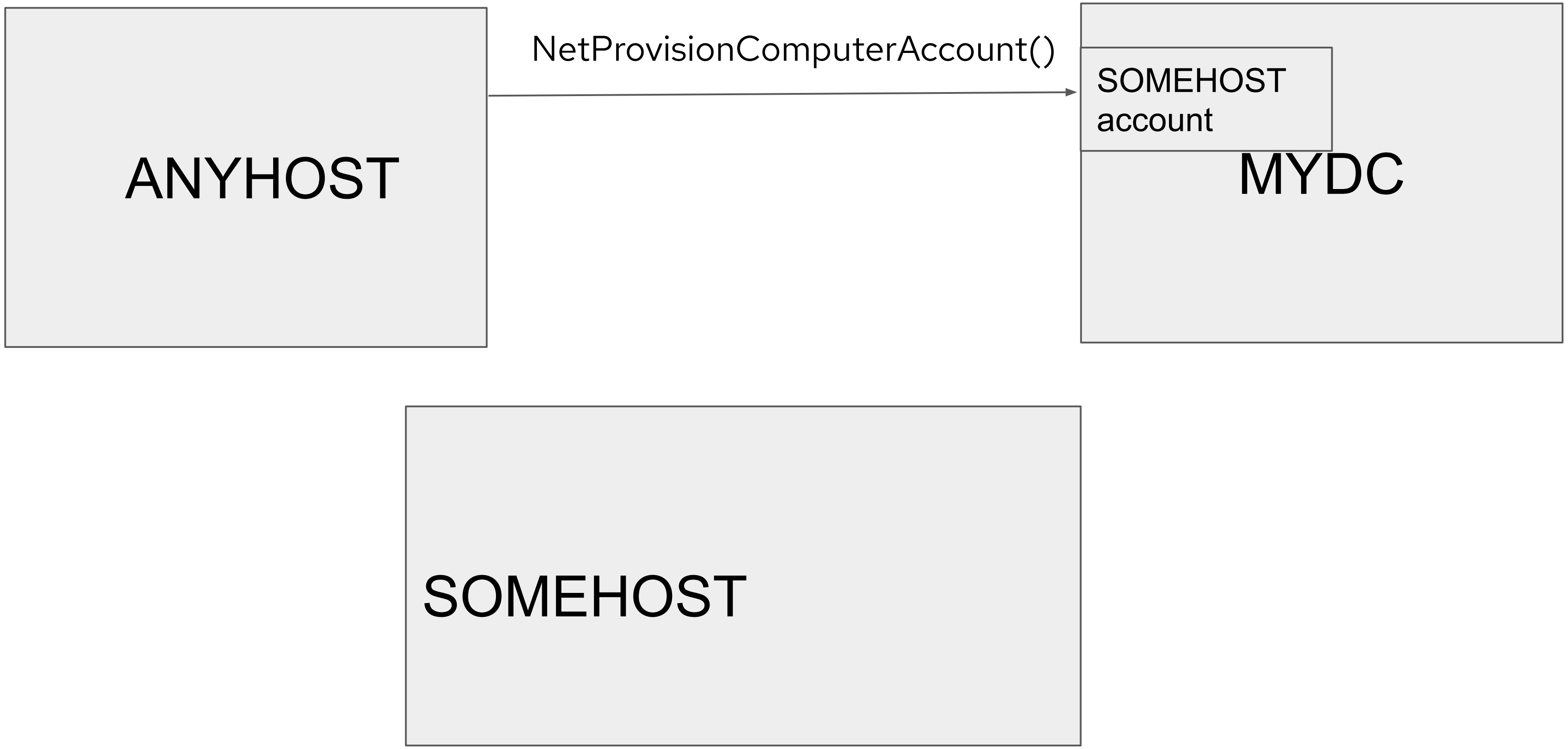
ANYHOST

MYDC

SOMEHOST

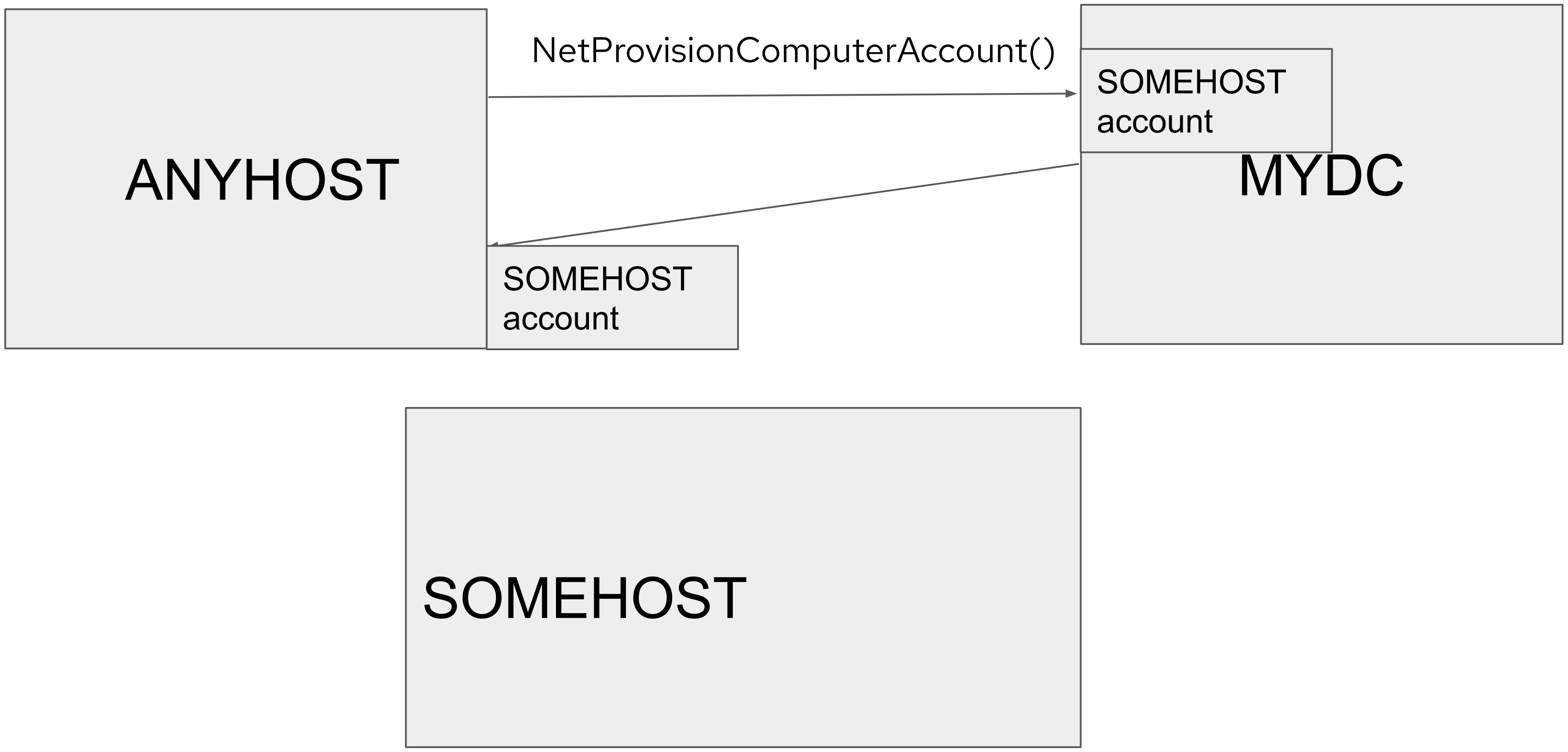
Offline Domain Join

Process - Provisioning



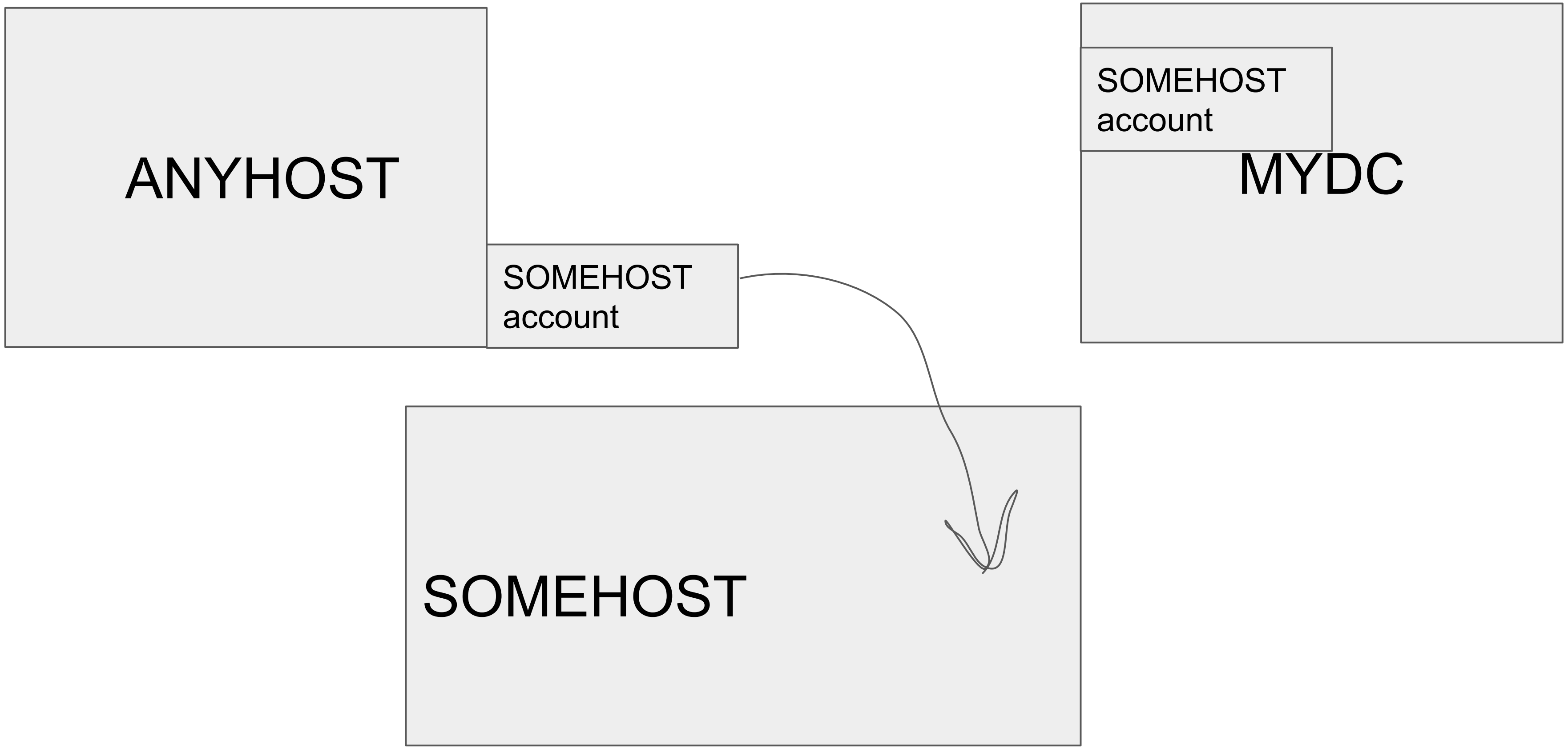
Offline Domain Join

Process - Provisioning



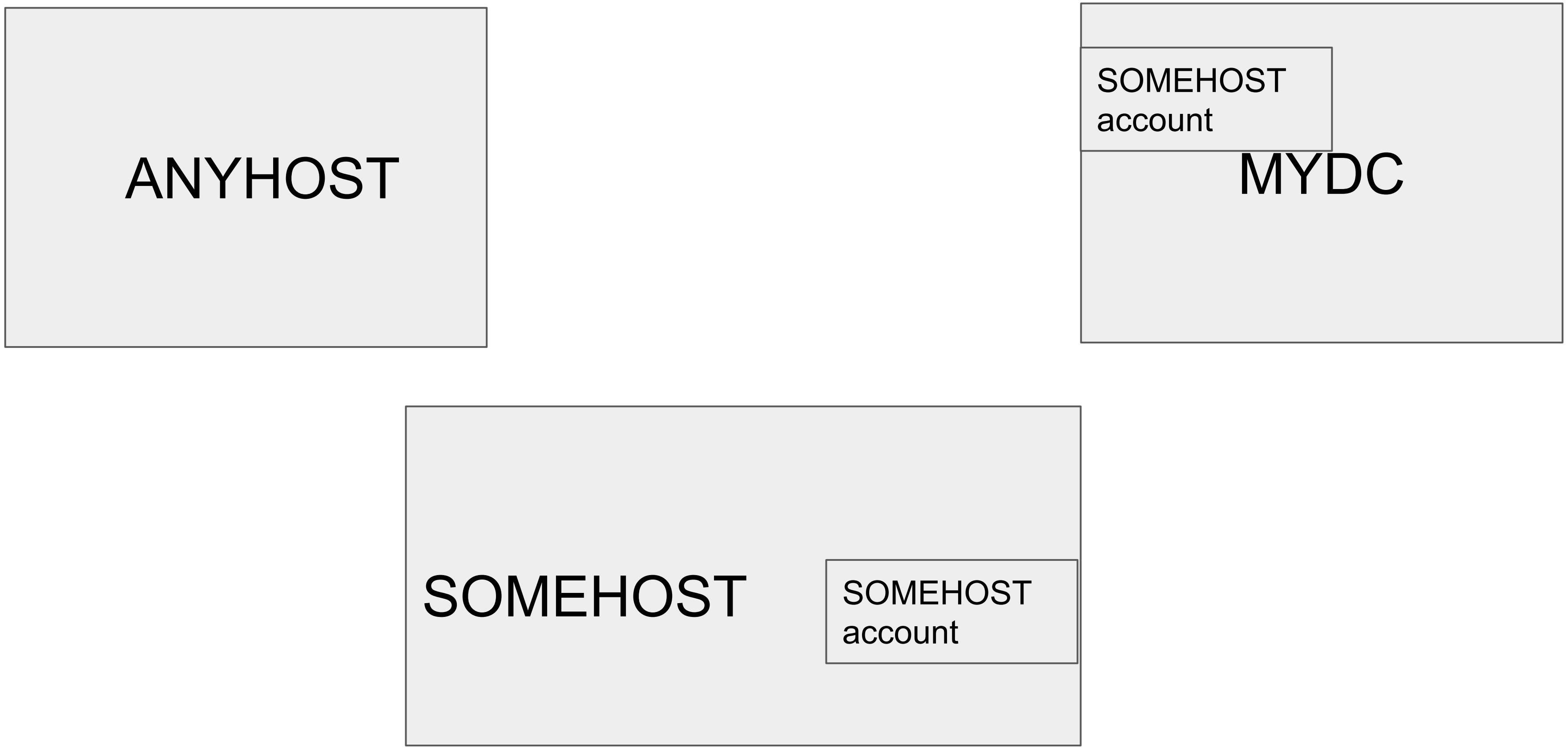
Offline Domain Join

Process - Request Offline Join



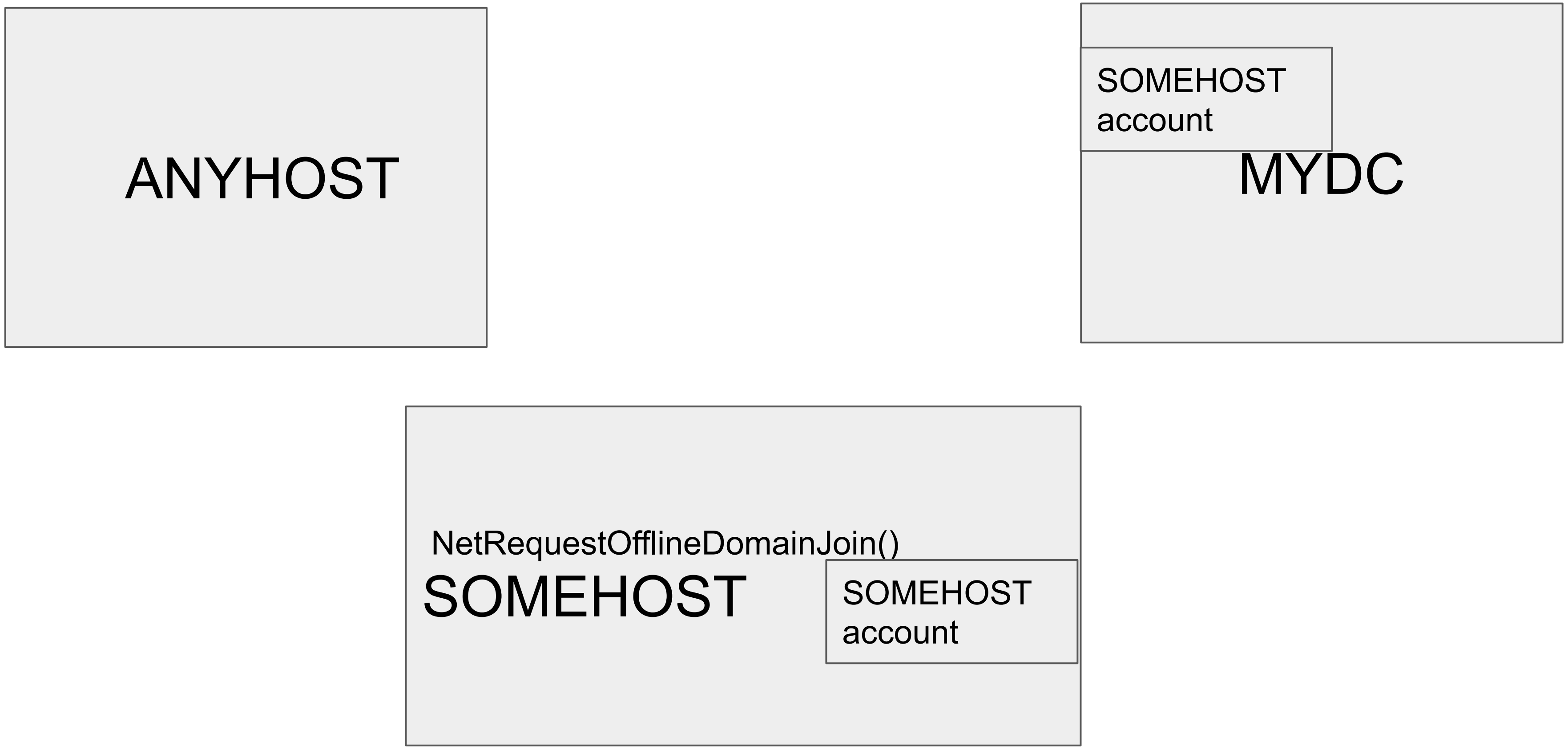
Offline Domain Join

Process - Request Offline Join



Offline Domain Join

Process - Request Offline Join



ANYHOST

SOMEHOST account

MYDC

NetRequestOfflineDomainJoin()

SOMEHOST

SOMEHOST account

Offline Domain Join (ODJ)

Exchange data format, what’s in there???

- Provisioning creates data indeed suited for interoperability
- Well documented, base64-encoded, NDR formatted, extensible struct
- ODJ structs typically exchanged via files

Offline Domain Join (ODJ)

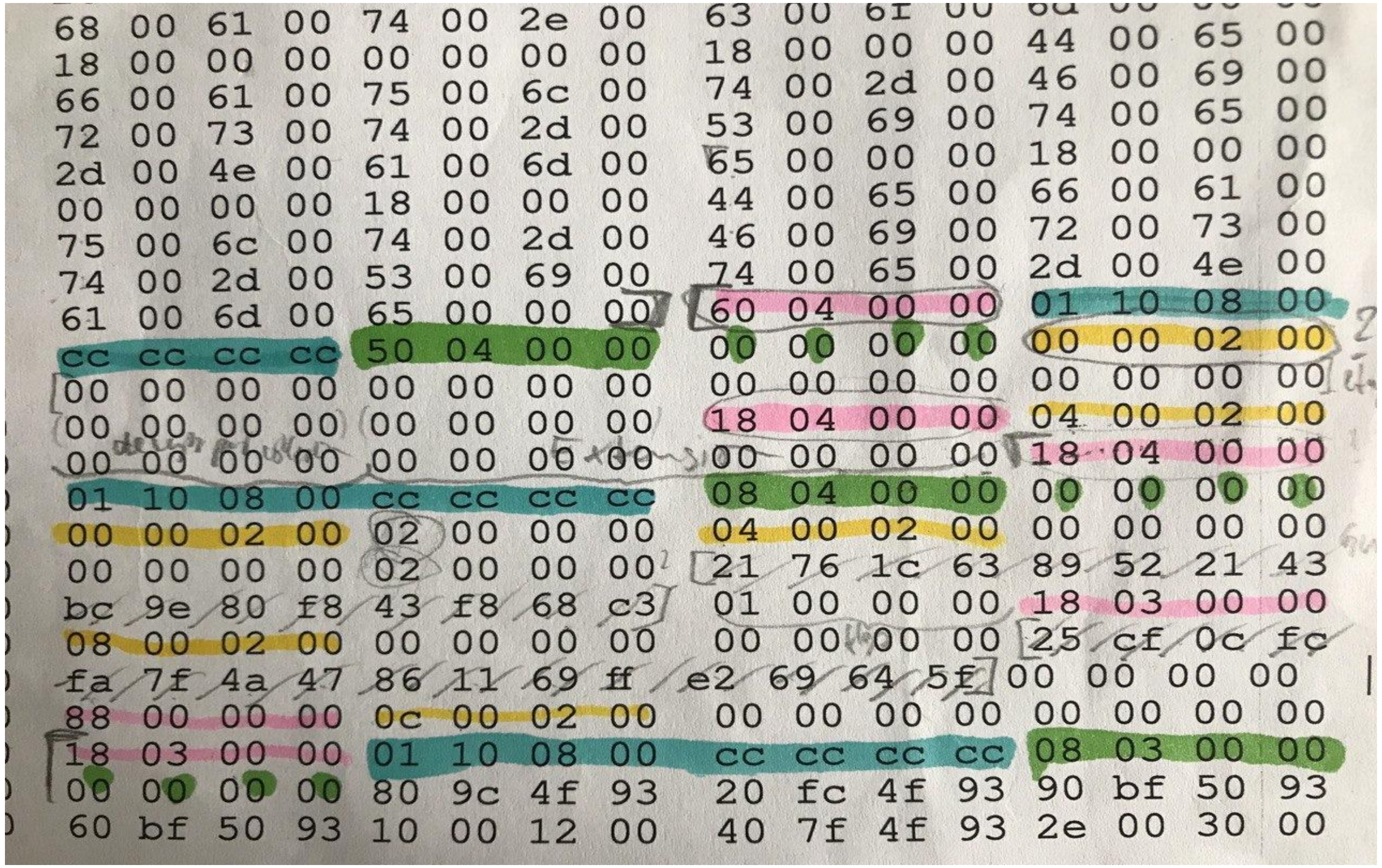
ODJ.idl

- IDL provided by Microsoft (Offline Domain Join IDL Definitions - Win32 apps)
- NDR formatted
- Excessive use of nested serialization stream pointers

  [MS-RPCE] 2.2.6 Type Serialization Version 1
- Pidl allows full autogeneration of these structures!
- Extensibility allows support for OS specific data:
  - `ODJ_WIN7_BLOB, OP_JOIN_PROV2_PART, OP_JOIN_PROV3_PART`
  - `OP_CERT_PART, OP_POLICY_PART`

Offline Domain Join (ODJ)

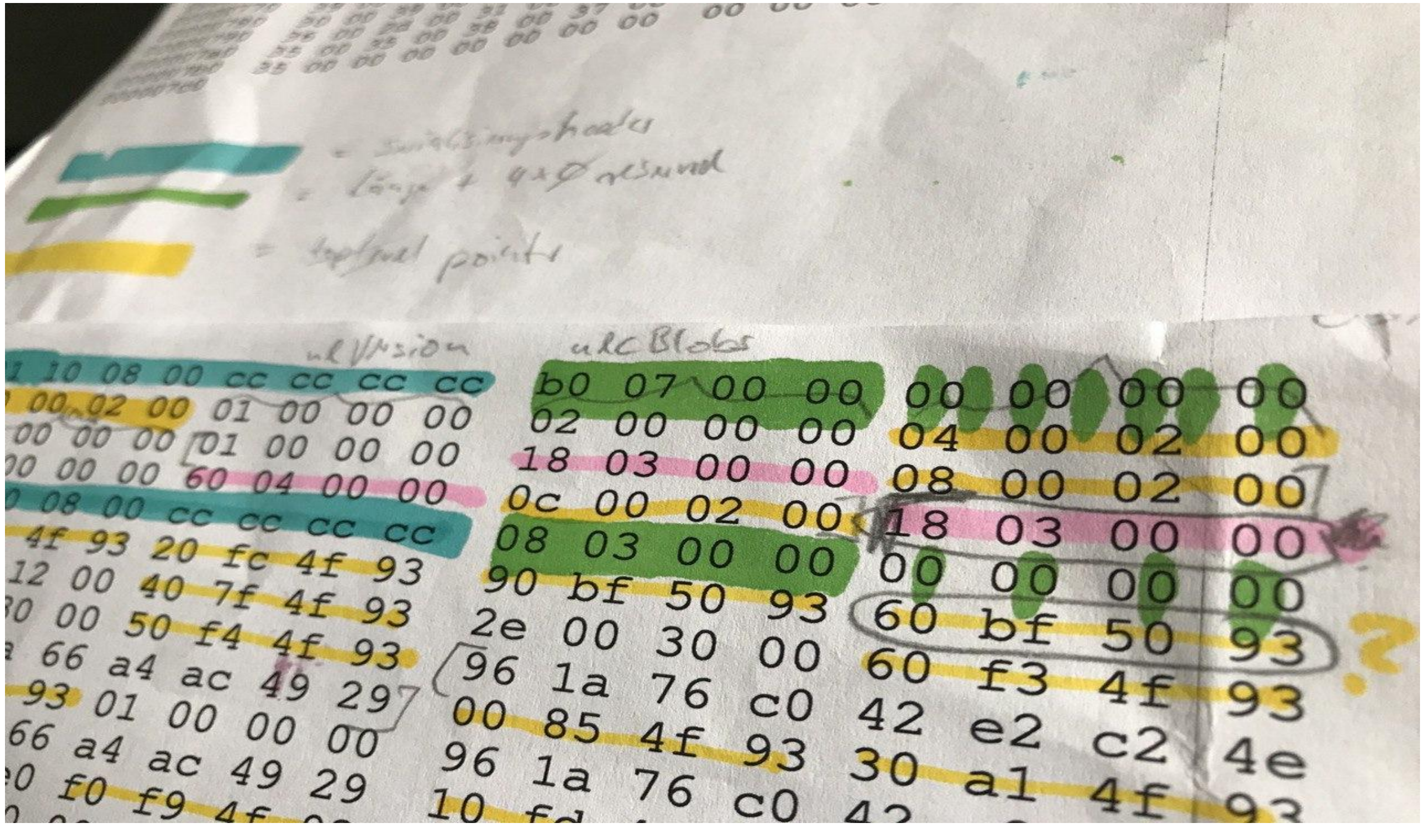
Nested serialization pointers? 😱



```
68 00 61 00 74 00 2e 00   63 00 6f 00 6d 00 00 00
18 00 00 00 00 00 00 00   18 00 00 00 44 00 65 00
66 00 61 00 75 00 6c 00   74 00 2d 00 46 00 69 00
72 00 73 00 74 00 2d 00   53 00 69 00 74 00 65 00
2d 00 4e 00 61 00 6d 00   65 00 00 00 18 00 00 00
00 00 00 00 18 00 00 00   44 00 65 00 66 00 61 00
75 00 6c 00 74 00 2d 00   46 00 69 00 72 00 73 00
74 00 2d 00 53 00 69 00   74 00 65 00 2d 00 4e 00
61 00 6d 00 65 00 00 00   60 04 00 00 01 10 08 00
cc cc cc cc 50 04 00 00   00 00 00 00 00 00 02 00
00 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00   18 04 00 00 04 00 02 00
00 00 00 00 00 00 00 00   00 00 00 00 18 04 00 00
01 10 08 00 cc cc cc cc   08 04 00 00 00 00 00 00
00 00 02 00 02 00 00 00   04 00 02 00 00 00 00 00
00 00 00 00 02 00 00 00   21 76 1c 63 89 52 21 43
bc 9e 80 f8 43 f8 68 c3   01 00 00 00 18 03 00 00
08 00 02 00 00 00 00 00   00 00 00 00 25 cf 0c fc
fa 7f 4a 47 86 11 69 ff   e2 69 64 5f 00 00 00 00
88 00 00 00 0c 00 02 00   00 00 00 00 00 00 00 00
18 03 00 00 01 10 08 00   cc cc cc cc 08 03 00 00
00 00 00 00 80 9c 4f 93   20 fc 4f 93 90 bf 50 93
60 bf 50 93 10 00 12 00   40 7f 4f 93 2e 00 30 00
```

Offline Domain Join (ODJ)

Is this “?” padding or a pointer artefact?



= Serialisierungsheader
= Länge + 4x0 reserved
= toplevel pointer

ulVersion

ulcBlobs

```
10 08 00 cc cc cc cc   b0 07 00 00   00 00 00 00
00 02 00 01 00 00 00   02 00 00 00   04 00 02 00
00 00 00 01 00 00 00   18 03 00 00   08 00 02 00
00 00 00 60 04 00 00   0c 00 02 00   18 03 00 00
08 00 cc cc cc cc      08 03 00 00   00 00 00 00
4f 93 20 fc 4f 93      90 bf 50 93   60 bf 50 93
12 00 40 7f 4f 93      2e 00 30 00   60 f3 4f 93
30 00 50 f4 4f 93      96 1a 76 c0   42 e2 c2 4e
66 a4 ac 49 29         00 85 4f 93   30 a1 4f 93
93 01 00 00 00         96 1a 76 c0   42
66 a4 ac 49 29         10 fd
e0 f0 f9 4f
```

?

Offline Domain Join (ODJ)

netapi calls in Windows

- Windows implements ODJ functions in netapi library (see lmjoin.h):
  - `NetProvisionComputerAccount()`
  - `NetRequestOfflineDomainJoin()`
  - `NetCreateProvisioningPackage()`
  - `NetRequestProvisioningPackageInstall()`

Offline Domain Join (ODJ)

NetProvisionComputerAccount()

```
NET_API_STATUS NET_API_FUNCTION NetProvisionComputerAccount(
  LPCWSTR lpDomain,
  LPCWSTR lpMachineName,
  LPCWSTR lpMachineAccountOU,
  LPCWSTR lpDcName,
  DWORD   dwOptions,
  PBYTE   *pProvisionBinData,
  DWORD   *pdwProvisionBinDataSize,
  LPWSTR  *pProvisionTextData
);
```

- Connects to DC (using admin credentials)
- Creates machine account for lpMachineName
- Returns Provision(Bin)Data

Offline Domain Join (ODJ)

NetRequestOfflineDomainJoin()

-
  ```
  NET_API_STATUS NET_API_FUNCTION NetRequestOfflineDomainJoin(
    BYTE    *pProvisionBinData,
    DWORD   cbProvisionBinDataSize,
    DWORD   dwOptions,
    LPCWSTR lpWindowsPath
  );
  ```
- Loads the provisioning data into the local OS
- No network or admin credentials required

Offline Domain Join (ODJ)

Windows djoin.exe tool

- Tool that calls these APIs:
  - djoin.exe
- djoin.exe /provision
  - Creates the AD object and returns the ODJ blob
  - Requires network and admin credentials
- djoin.exe /requestodj
  - Takes the ODJ blob and applies it to the OS
  - Does not require network
  - OS does not even have to be started

Offline Domain Join (ODJ)

djoin.exe output

- Djoin on Windows creates and consumes UTF16/UCS2 encoded blob of base64 encoded data of NDR formatted structures (with heavily using serialized data streams)
- This data can be embedded in Unattended.xml files, providing unattended join (using Windows setup files)
- Cryptographic data (machine account password) passed in clear (!)

Offline Domain Join (ODJ)

Security Note

“Security Note: The blob returned by the NetProvisionComputerAccount function contains very sensitive data. It should be treated just as securely as a plaintext password. The blob contains the machine account password and other information about the domain, including the domain name, the name of a domain controller, and the security ID (SID) of the domain. If the blob is being transported physically or over the network, care must be taken to transport it securely. The design makes no provisions for securing this data. This problem exists today with unattended setup answer files which can carry a number of secrets including domain user passwords. The caller must secure the blob and the unattended setup files. Solutions to this problem are varied. As an example, a pre-exchanged key could be used to encrypt a session between the consumer and provisioning entity enabling a secure transfer of the opaque blob.”

https://docs.microsoft.com/en-gb/windows/win32/api/lmjoin/nf-lmjoin-netprovisioncomputeraccount

Offline Domain Join

Mechanism and tools in Samba

libnetjoin interface

- Central interface for domain joining
- Used by “net”, “libnetapi” and workstation DCE/RPC services
- Provides all hooks for offline joining
- Consumes and emits ODJ structures
- Recent addition:
    - python wrapper for libnet_Join() and libnet_Unjoin() to call libnetjoin from samba-tool

libnetapi interface

- Implements the ~~4~~2 relevant API calls for offline joining
  - NetProvisionComputerAccount()
  - NetRequestOfflineDomainJoin()
  - ~~NetCreateProvisioningProvisioningPackage()~~
  - ~~NetRequestProvisioningPackageInstall()~~
- Comes with example tools including a `djoin.exe` clone that provides same command line experience as in Windows

“net offlinejoin”

- Calls libnetapi from the samba3 net binary
- Basically provides djoin.exe style commands and options
- “net offlinejoin provision help”
- “net offlinejoin requestodj help”

Demo

“net offlinejoin” joining a Samba client to AD

Demo

“net offlinejoin” joining a Windows 10 client to AD

Demo

"djoin.exe" on Windows 10 joining Samba client to AD

What’s next?

Use cases of ODJ in Samba

- Offload machine account creation completely to AD admins
- Samba could consume ODJ blobs for (mass-)deployment of joined Samba clients in containerized environments
  - John Mulligan / Michael Adam talk on the Samba operator tomorrow!

Next steps

- Generation of Unattended.xml files to allow unattended join for Windows clients
- Discuss whether to extend the API calls (e.g. also allow backup of domain join metadata structures in secrets.tdb)
- Full support for Group Policies and Certificate deployment for Linux
- Define our own structures for deployment of samba specific content

Further reading

The docs, the docs, all the answers are in the docs!

- Microsoft:
  - Offline Domain Join (Djoin.exe) Step-by-Step Guide
  - NetProvisionComputerAccount function
  - NetRequestOfflineDomainJoin function
  - Offline Domain Join IDL Definitions
  - MS-RPCE 2.2.6 Type Serialization Version 1
- Samba:
  - git.samba.org WIP branch
  - https://gitlab.com/samba-team/samba/-/merge_requests/1943

Thank you!

- linkedin.com/company/red-hat
- youtube.com/user/RedHatVideos
- facebook.com/redhatinc
- twitter.com/RedHat