CANONICAL

Integration of Ubuntu Desktop with Microsoft Active Directory

June 2019

Executive Summary

Microsoft Active Directory is a widely-deployed directory service that is commonly used for identity management and authentication across the enterprise. Its ubiquity makes it a fixture of the IT landscape. As a result, interoperability with Active Directory is often a necessity when deploying services based on non-Microsoft operating systems.

Powerbroker Identity Service Open (PBIS Open) is one tool whose purpose is precisely to ease integration of non-Microsoft operating systems into an existing Active Directory architecture. PBIS Open automates a number of settings that previously required time-consuming and error-prone manual configuration, making integration of Ubuntu with Active Directory quick and painless as it takes the guesswork out of the process. Additionally, Open-Source Active Directory Bridging, a proprietary product made by BeyondTrust and from which PBIS Open is derived, provides a set of tools to enable the use of group policies with Linux (including Ubuntu).

PBIS Open also supports disconnected operations where a computer which was connected to the corporate network is then removed (e.g. a laptop on an aeroplane), bringing network directory access to laptop users.

It is important to understand that Active Directory was never meant to be a cross-platform directory service in the first place. From the ground up, it was built with Microsoft operating systems and software in mind. As such, complete Active Directory integration of third-party operating systems, such as Ubuntu, is not straight forward. The identity management component of Active Directory (authentication of users and groups) is open enough for tools such as PBIS Open to achieve a functional level of interoperability. However, do not assume further Active Directory tasks, such as system management and provisioning for example, are interoperable with Ubuntu.

Introduction

Overview

This whitepaper discusses the use of Powerbroker Identity Service Open on Ubuntu 18.04 LTS. We will demonstrate how an Ubuntu desktop can be configured to enable Active Directory users to login using Powerbroker Identity Service Open.

The section 'Overview of Powerbroker Identity Service Open' explains what Powerbroker Identity Service is, and briefly presents its architecture with an eye toward understanding how it fulfills its functions in Ubuntu.

Pre-requisites and configuration checklists for both Active Directory and Ubuntu members are discussed in the 'Active Directory setup checklist' and 'Ubuntu setup checklist', respectively. This section is not specific to our example setup; it holds true for any deployment of Powerbroker Identity Service Open, and can be used as a starting point of your own.

The section 'Integrating Ubuntu with Active Directory' presents the steps required to connect an Ubuntu desktop machine. At the end of this section, you will be able to log in to your Ubuntu machine using Active Directory credentials.

Finally, we present a number of alternatives to Powerbroker Identity Service Open and discuss their advantages and drawbacks.

Intended audience

This whitepaper has been written with Windows system administrators new to Ubuntu in mind. We assume a basic level of knowledge of administering Ubuntu, including the ability to use a command-line shell, understanding of sudo as a means for privilege escalation and the ability to use a text editor to edit configuration files.

Overview of Powerbroker Identity Service Open

The purpose of Powerbroker Identity Service Open is to simplify integration of non-Microsoft systems (Linux and Mac OS) into Microsoft Active Directory. It is based in some part on a component of the Samba open source project called winbind. The purpose of winbind is to act as a gateway to Microsoft domains for authentication and identity resolution of users, and to provide consistent mapping of users and groups. It basically enables Microsoft domain users and groups to appear to be local on the non-Microsoft system. This is very useful in a number of scenarios, particularly when sharing files between Windows and Linux using Samba.

To make full use of winbind as a source of local users and groups on Linux requires a fair bit of effort. The Name Service Switch (NSS) framework needs to be configured to resolve users and groups against winbind. The Pluggable Authentication Module (PAM) stack similarly needs to be configured to funnel authentication requests through winbind. The behaviour of winbind itself is configurable to a large extent, and getting the intended result may involve tweaking its configuration considerably. Powerbroker Identity Service Open consolidates all these operations in a single tool and delivers a clean configuration for the common use-case in a few easy steps.

Powerbroker Identity Service Open, through a PAM module, provides a generic mechanism for system services on Ubuntu to validate user's credentials against Active Directory. This would allow, for example, a mail server, a web service, or any other application that supports the PAM framework, to authenticate users belonging to the Active Directory the Ubuntu server is joined to. This pre-empts the need to keep multiple redundant authentication databases by centralising user accounts management, and enables organisations to make full use of their existing Active Directory infrastructure and know-how when deploying Ubuntu.

Once Powerbroker Identity Service Open is installed and configured, users from the Active Directory will appear as if they are local to the Ubuntu system. User attributes that are standard in Unix/Linux but not present in Active Directory are either generated algorithmically on the fly (i.e; the numerical user ID), or through configuration directive (home directory location and preferred user shell.) In the same manner, groups from the Active Directory will also appear to be regular Unix groups. This is achieved through a Name Service Switch (NSS) module, a mechanism that is standard across all Linux distributions.

Powerbroker Identity Service Open provides a command-line tool to manage the services. The command-line tool, domainjoin-cli, makes joining an Active Directory domain very straightforward and provides various sanity checks on the Ubuntu computer's configuration - more on that later. Under the hood, Powerbroker Identity Service Open is running a daemon (a long-running system service) called lwsmd. The PBIS Service Manager (lwsmd) provides a service control architecture for starting and stopping all PBIS daemons and drivers based on a dependency graph. The lwsmd daemon itself is managed using a standard SysV init script.

Advantages of PowerBroker Identity Service Open

- No software to install on the Active Directory, and no change to its configuration required.
- Centralised authentication use for existing user and group when deploying Ubuntu, no need to maintain duplicate user database.
- Unix user and group ID are coherent across all machines running Powerbroker Identity Service Open, no need to maintain an ID map.
- Disconnected operation enables mobile users to authenticate using their Active Directory credentials.

Drawbacks

- No control over assignment of Unix user and group ID; they are computed algorithmically by Powerbroker Identity Service Open.
- No fine-grained control over which Active Directory users and groups are exposed by Powerbroker Identity Service Open client.
- Integration limited to identity management and authentication. However, please see the subsection on Powerbroker Identity Service Enterprise at the end of this white paper for information on a more feature-full alternative.

Setting up Active Directory

On the Active Directory server you have to verify that you have an account with administrative privileges, that the Active Directory service and the DNS are configured and operational, and that the user accounts have been added to the directory.

Administrative Privileges

You will need an account with sufficient privileges to add the Ubuntu computers to the Active Directory. Typically, this would be an account member of the Domain Administrator group (such as the ubiquitous Administrator account), although this can vary according to your Active Directory configuration.

AD and DNS

Active Directory and a DNS must be installed, configured and running on your domain controller. Depending on your network configuration, the DNS can run on a separate server but you will need to ensure that a DNS entry has been created for the machine in question.

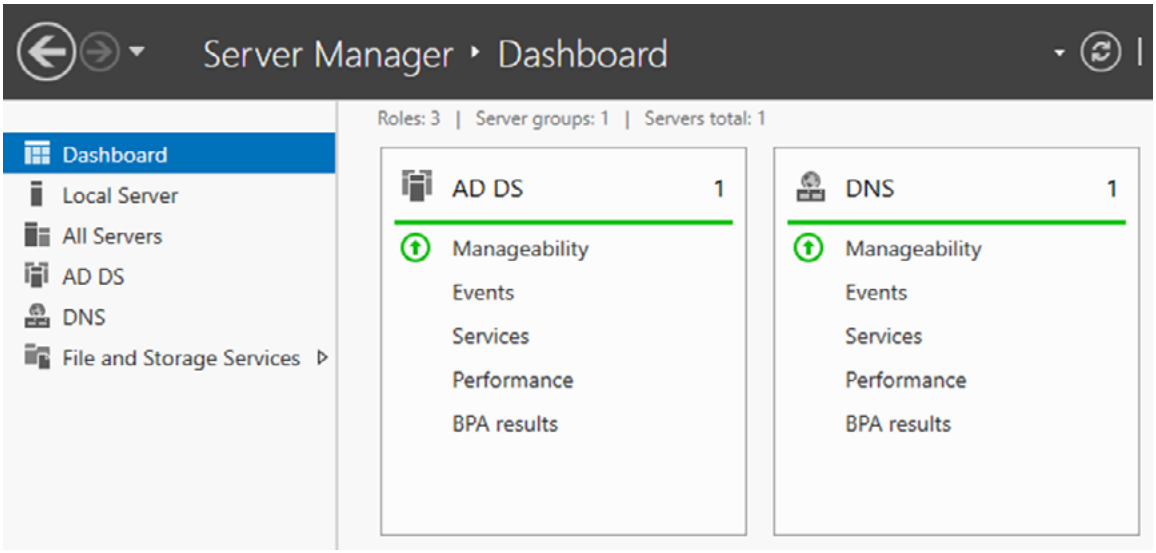


Image 1: Windows Server 2016 Dashboard

While not strictly a requirement, it is better to have a reverse lookup zone configured - containing pointer records (PTR) - for your domain in the Active Directory DNS, as many services in Linux do make use of reverse lookup. From the Ubuntu command line, you can easily check if the reverse lookups have been configured properly by using the dig or the host command.

Firstly , verify that the hostname can be resolved:

```
ubuntu@ad-desktop-1:~$ host ad-desktop-1.warthogs.biz
ad-desktop-1.warthogs.biz has address 10.2.250.153
```

Then that the reverse lookup is configured properly with host:

```
ubuntu@ad-desktop-1:~$ host 10.2.250.153
153.250.2.10.in-addr.arpa domain name pointer ad-desktop-1.warthogs.
biz.
```

Or dig:

```
ubuntu@ad-desktop-1:~$ dig  -x 10.2.250.153

; <<>> DiG 9.11.3-1ubuntu1.3-Ubuntu <<>> -x 10.2.250.153
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37501
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;153.250.2.10.in-addr.arpa.      IN
PTR

;; ANSWER SECTION:
153.250.2.10.in-addr.arpa. 0    IN
PTR                             ad-desktop-1.warthogs.biz.

;; Query time: 0 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Thu Dec 20 04:22:07 EST 2018
;; MSG SIZE  rcvd: 93
```

Organisational Units

If your Active Directory domain is divided into organisational units (OU), you will need to determine into which OUs you want to join the Ubuntu computers. In this whitepaper, for the sake of simplicity, the structure of the directory will be flat and we will not make use of OUs.

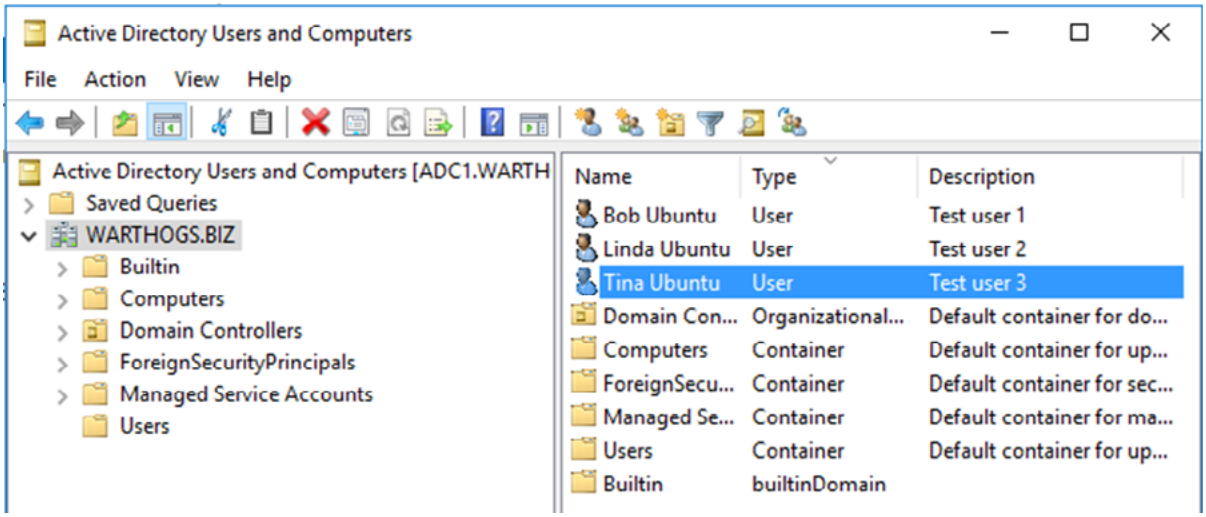


Image 2: Example set up of Ubuntu computers

Setting up Ubuntu Desktop

On Ubuntu's side, besides a correct network setup and a user with administrative privileges, you'll need to ensure that time is synchronised with the Active Directory controller and host name resolution is working.

Administrative privileges

As is usual in Ubuntu, all examples of command requiring super-user (administrative) privileges in the text have been prefixed with sudo. You are not expected to have access to the root account (it is disabled by default on Ubuntu), but you are expected to have a user account member of the admin group, who is allowed to escalate privileges using the sudo command. The first user account, created during installation, is a member of the admin group in question. In our case, this user is called, quite simply, 'ubuntu'.

Network settings

Using Powerbroker Identity Service Open obviously requires network connectivity to the Active Domain Controller. While it is possible to use Powerbroker Identity Service Open on a machine configured for dynamic IP addressing using DHCP, our example will assume fixed IP settings with a single network interface for the sake of simplicity.

If a firewall is mitigating IP connectivity between the Ubuntu machine and the ADC, you will need to ensure that the required ports are open for connection between the ADC and the Ubuntu machines. Please refer to the 'Powerbroker Identity Service Open Installation and Administration Guide' (see the 'Further reading' section at the end of this document) for the complete list of ports required. Powerbroker Identity Service Open itself is not listening on any port for an inbound connection; as such, no change will need to be made if you are using a host-based firewall, such as iptables or ufw, on the Ubuntu machine.

Host name resolution

It is important to ensure that the fully qualified domain name (FQDN) of the Ubuntu machine matches the DNS record used in the Active Directory DNS. This information is stored in the /etc/hostname configuration file on Ubuntu, and a matching entry must exist in /etc/hosts. You can check the FQDN of the Ubuntu machine from the command line using the hostname command, for example:

```
ubuntu@ad-desktop-1:~$ hostname -f
ad-desktop-1.warthogs.biz
```

Time synchronisation

The Kerberos protocol, used internally by Active Directory for authentication, is sensitive to clock skew between computers participating in a Kerberos domain. The default clock skew tolerance is 300 seconds (five minutes). If the Ubuntu machine and the ADC clock drift apart for more than five minutes, authentication against the ADC will systematically fail.

Traditionally, in the Unix/Linux world, time synchronisation is achieved using the Network Time Protocol (NTP). This is usually completed against an external time source, such as one of the many public NTP servers on the internet. By default, Ubuntu is configured to synchronise time with the ntp.ubuntu.com NTP server each time a network interface is brought up, which happens at least at every boot.

In our case, it is not desirable to have the Ubuntu machine synchronise time with an outside source, as this source may differ from the ADC. Hence, the default NTP server needs to be changed to one of the ADC. Since Ubuntu 16.04 LTS, Ubuntu, by default, uses timedatectl / timesyncd (which are part of systemd). It replaces most of ntpdate / ntp.

The current status of time and time configuration via *timedatectl* and *timesyncd* can be checked with timedatectl status:

```
# timedatectl status
                  Local time: Thu 2018-12-20 09:40:54 EST
              Universal time: Thu 2018-12-20 14:40:54 UTC
                    RTC time: Thu 2018-12-20 14:40:55
                   Time zone: America/New_York (EST, -0500)
     System clock synchronized: yes
systemd-timesyncd.service active: yes
             RTC in local TZ: no
```

Via *timedatectl* an admin can control the timezone, how the system clock should relate to the hwclock and if permanent synchronisation should be enabled or not. See man timedatectl for more details.

The nameserver to fetch time for *timedatectl* and *timesyncd* from can be specified in /etc/systemd/timesyncd.conf and additional config files can be stored in /etc/systemd/timesyncd.conf.d/. The entries for NTP= and FallbackNTP= are space separated lists. See man timesyncd.conf for more.

timesyncd will generally do the right thing keeping your time in sync, and *chrony* will help with more complex cases.

The content of timesyncd.conf for our example setup is pasted below:

```
#  This file is part of systemd.
#
#  systemd is free software; you can redistribute it and/or modify it
#  under the terms of the GNU Lesser General Public License as
published by
#  the Free Software Foundation; either version 2.1 of the License, or
#  (at your option) any later version.
#
# Entries in this file show the compile time defaults.
# You can change settings by editing this file.
# Defaults can be restored by simply deleting this file.
#
# See timesyncd.conf(5) for details.

[Time]
NTP=adc1.warthogs.biz
FallbackNTP=ntp.ubuntu.com
#RootDistanceMaxSec=5
#PollIntervalMinSec=32
#PollIntervalMaxSec=2048
```

Setting up PowerBroker Identity Service Open

Installing the package

The first task is to download PowerBroker Identity Service Open package from the upstream repository.

To download PowerBroker Identity Services Open “AD Bridge”, please visit the release page on github and select latest version for Ubuntu. For this whitepaper we used the package pbis-open-8.8.0.506.linux.x86_64.deb.sh.

Then open a terminal (from an Ubuntu Desktop press CTRL+ALT+T or Super then type terminal and select the terminal application) and run the command:

```
$ sudo sh ./pbis-open-8.8.0.506.linux.x86_64.deb.sh
```

If all goes well it will return without any error. You can verify that it is correctly installed with the standard apt command:

```
$ apt policy pbis-open
pbis-open:
  Installed: 8.8.0.506
  Candidate: 8.8.0.506
  Version table:
 *** 8.8.0.506 100
        100 /var/lib/dpkg/status
```

Removing the package

If you want to remove the machine from the domain and remove the un-needed package pbis-open, you can do it the usual way from the command line:

```
$ sudo apt autoremove --purge pbis-open pbis-open-upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  pbis-open* pbis-open-upgrade*
0 upgraded, 0 newly installed, 2 to remove and 1 not upgraded.
After this operation, 22.3 MB disk space will be freed.
Do you want to continue? [Y/n]
```

Connecting Ubuntu to Active Directory

Joining a domain

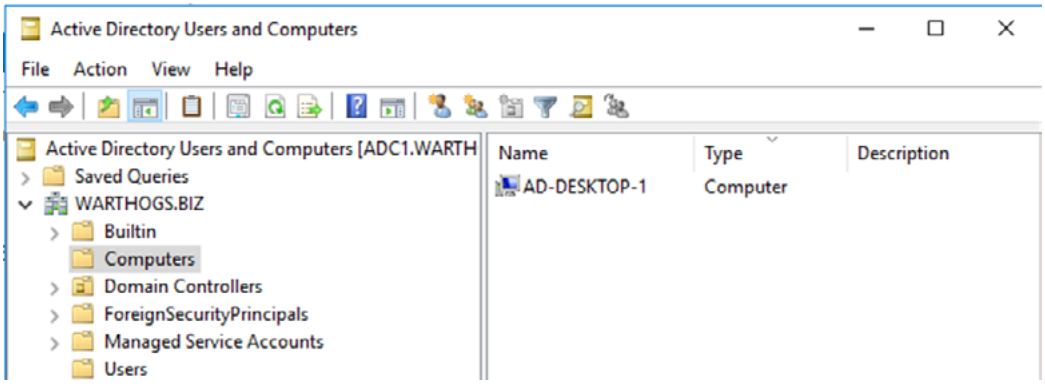
Once the PowerBroker Identity Service Open package is installed, you can proceed with joining the domain using the domainjoin-cli command-line tool, for example:

```
# sudo domainjoin-cli join warthogs.biz Administrator
Joining to AD Domain:
warthogs.biz
With Computer DNS Name: ubuntusrv1.warthogs.biz
Administrator@WARTHOGS.BIZ's password: *****
SUCCESS
```

You can substitute 'warthogs.biz' for your own domain name, and 'Administrator' for an account with sufficient privileges to join computers in your domain.

Verifying proper domain operations

To verify that Powerbroker Identity Service Open works as intended, you may want to check that the Ubuntu server is listed in the Active Directory Users and Computers MMC snap-in.



On the Ubuntu computer, confirm that it has indeed joined the domain by querying PowerBroker Identity Service Open, for example:

```
ubuntu@ad-desktop-1:~$ sudo domainjoin-cli query
[sudo] password for ubuntu:
Name = ad-desktop-1
Domain = WARTHOGS.BIZ
Distinguished Name = CN=AD-DESKTOP-1,CN=Computers,DC=WARTH
OGS,DC=BIZ
```

Next, verify that users from the domain can be resolved using the getent command, for example:

```
ubuntu@ad-desktop-1:~$ getent passwd WARTHOGS\\linda
WARTHOGS\linda:PBIS:1515717715:1515717121:Linda Ubuntu:/home/
local/WARTHOGS/linda:/bin/sh
```

The getent command is used to query NSS databases. In the above case, we ask getent to query the password (passwd) database for the 'WARTHOGS\linda' entry. The entry format is the same as is used in the /etc/passwd system user database, except the entry is not actually from /etc/passwd; it is pulled from the Active Directory. You may also use getent to resolve a group entry, as below:

```
ubuntu@ad-desktop-1:~$ getent group WARTHOGS\\domain^users
WARTHOGS\domain^users:PBIS:1515717121:WARTHOGS\
linda,WARTHOGS\tina
ubuntu@ad-desktop-1:~$
ubuntu@ad-desktop-1:~$ getent group WARTHOGS\\marketing
WARTHOGS\marketing:PBIS:1515717719:WARTHOGS\
linda,WARTHOGS\bob
```

Notice that both users and groups from the Active Directory are pre-fixed with the domain name according to the usual DOMAIN\ convention.

> The backslashes "\" have a special meaning as the so-called 'escape' character in Unix shells. As such, when you use the DOMAIN\user convention at a shell prompt, the backslash will need to be doubled to prevent escaping, for example DOMAIN\\user. Alternatively, you can quote the expression containing a backslash, as with 'DOMAIN\user'.

Powerbroker Identity Service also provides its own set of commands to list users and groups. These commands are located in the directory /opt/pbis/bin/enum-users : lists the users registered in Active Directory

```
root@ad-desktop-1:~# /opt/pbis/bin/enum-users^C
User info (Level-0):
====================
Name:          WARTHOGS\administrator
Uid:           1515717108
Gid:           1515717121
Gecos:         <null>
Shell:         /bin/sh
Home dir:      /home/local/WARTHOGS/administrator

[...]

User info (Level-0):
====================
Name:          WARTHOGS\louise
Uid:           1515717721
Gid:           1515717121
Gecos:         Louise Ubuntu
Shell:         /bin/sh
Home dir:      /home/local/WARTHOGS/louise

TotalNumUsersFound: 9
```

enum-groups : lists the groups registered in Active Directory

```
root@ad-desktop-1:~# /opt/pbis/bin/enum-groups^C
root@ad-desktop-1:~# cat /tmp/groups
Group info (Level-0):
====================
Name: WARTHOGS\domain^computers
Gid:  1515717123
SID:  S-1-5-21-1489332398-964710433-1994175689-515

[...]

Group info (Level-0):
====================
Name: WARTHOGS\engineering
Gid:  1515717718
SID:  S-1-5-21-1489332398-964710433-1994175689-1110

Group info (Level-0):
====================
Name: WARTHOGS\marketing
Gid:  1515717719
SID:  S-1-5-21-1489332398-964710433-1994175689-1111

TotalNumGroupsFound: 22
```

In this whitepaper, we won't cover the other commands. These commands are documented in PowerBroker Identity Service reference documentation.

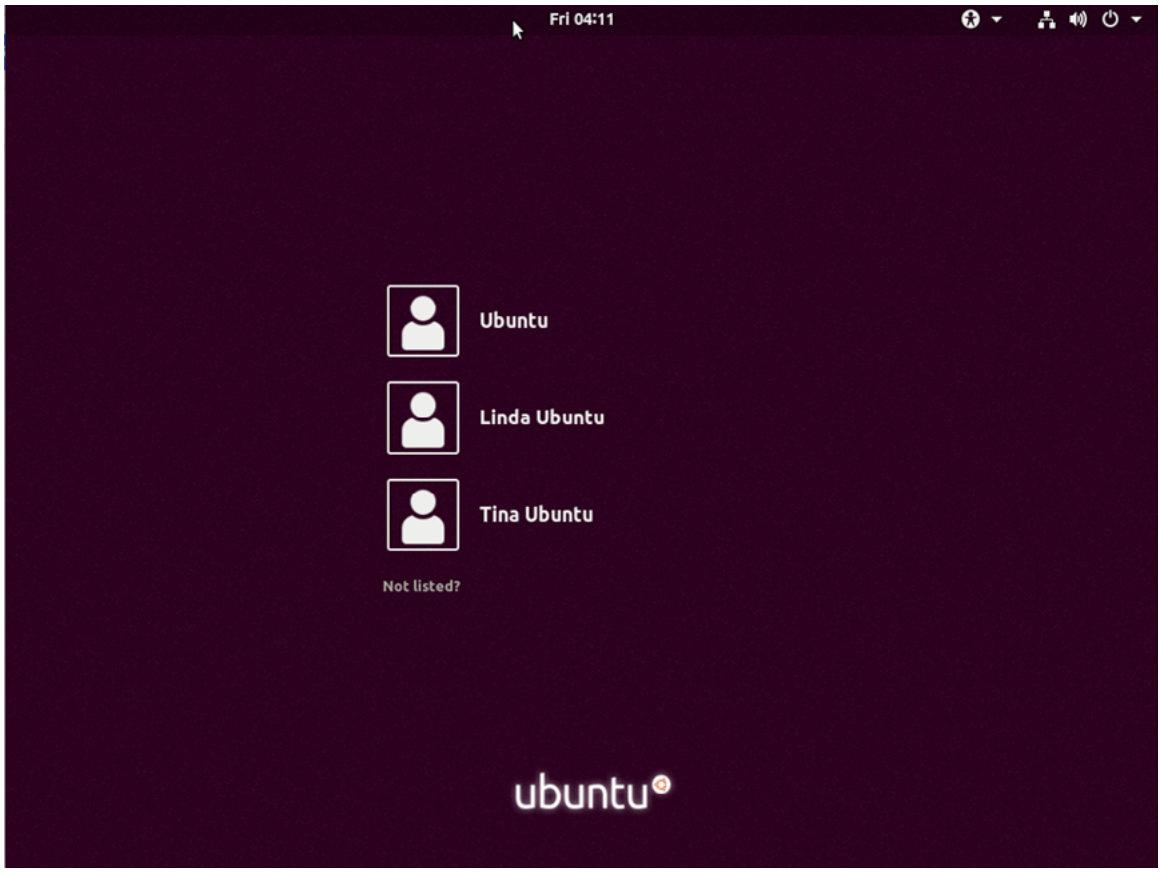
Next, you can test authentication by using SSH to connect to your Ubuntu server. As we already have a shell running on the Ubuntu server, we can simply open an SSH connection to 'localhost' using an Active Directory user, for example:

```
ubuntu@ad-desktop-1:~$ ssh WARTHOGS\\linda@ad-desktop-1.
warthogs.biz
Password:
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-42-generic x86_64)
 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch
9 packages can be updated.
8 updates are security updates.
Last login: Wed Dec 19 08:56:59 2018 from 10.2.250.153
$
```

From the above, we can see that we were able to establish an SSH connection using the credentials of an Active Directory user, confirming the ability to authenticate to the Active Directory. The 'whoami' and 'id' commands further confirm that resolving users and groups work as expected. We have finished testing the installation.

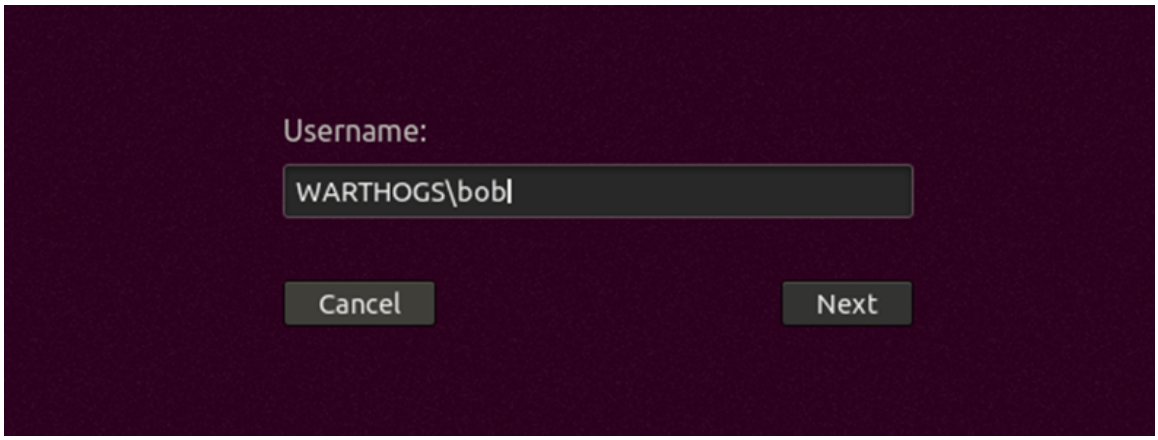
```
$ whoami
WARTHOGS\linda
$
$ id
uid=1515717715(WARTHOGS\linda) gid=1515717121(WARTHOGS\
domain^users) groups=1515717121(WARTHOGS\
domain^users),1515717719(WARTHOGS\marketing)
$
```

Lastly, you can verify that a user can authenticate with a graphical interface from GDM. On the login screen the list of users includes the local users, in our example 'Ubuntu', and the list of Active Directory users which have already logged in at least once, in our example, 'Linda Ubuntu' and 'Tina Ubuntu'.

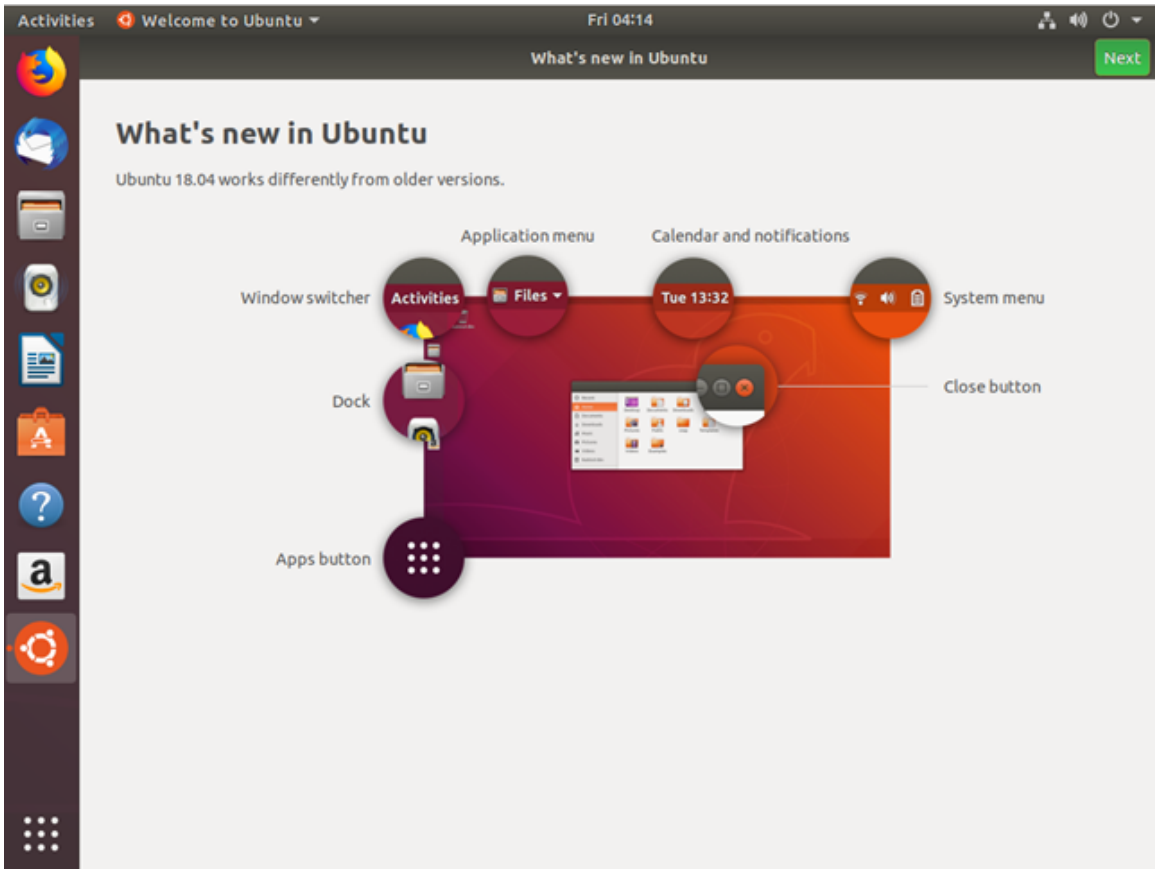


User 'Bob' is not listed and wants to log into this machine. He selects 'Not listed?' to display a login prompt and enter his username prefixed by the domain name.

Note that the separator is a single backslash because we are not in a Unix shell and it doesn't have to be escaped.



Upon entering his password, he successfully starts his session.



Additional configuration

Allowing sudo access to the admin group

You can now allow the administrator of the Active Directory controller to have administrator privileges on the machines controlled by this controller. In Ubuntu, it means all privileges to run sudo commands to the group administrator.

For instance, if we want to grant admin privileges on the Ubuntu machines to the group of user 'Administrator', the first thing is to determine the group of this user. This is done with the command list-group-for-user:

```
$ /opt/pbis/bin/list-groups-for-user WARTHOGS\\administrator
Number of groups found for user 'WARTHOGS\administrator' : 6
Group[1 of 6] name = WARTHOGS\
denied^rodc^password^replication^group (gid = 1515717180)
Group[2 of 6] name = WARTHOGS\domain^admins (gid = 1515717120)
Group[3 of 6] name = WARTHOGS\domain^users (gid = 1515717121)
Group[4 of 6] name = WARTHOGS\schema^admins (gid = 1515717126)
Group[5 of 6] name = WARTHOGS\group^policy^creator^owners (gid =
1515717128)
Group[6 of 6] name = WARTHOGS\enterprise^admins (gid =
1515717127)
```

The command returns 6 records. In our example, the Active Directory admin group we are interested in is WARTHOGS\domain^admins. Granting sudo access to the group is accomplished adding the group to /etc/sudoers and allowing it to run any command.

```
root@ad-desktop-1:~# visudo
[the default editor opens /etc/sudoers]

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL
%WARTHOGS\\domain^admins ALL=(ALL) ALL
```

Note the syntax with the domain and the double backslashes.

Now the administrator of the Active Directory controller can run any command with sudo.

```
 ubuntu@ad-desktop-1:~$ sudo -i -u WARTHOGS\\administrator
[sudo] password for ubuntu:
$ whoami
WARTHOGS\administrator
$ sudo ls
[sudo] password for WARTHOGS\administrator:
examples.desktop
$
```

Alternatively, you could create a specific group in Active Directory, eg UbuntuAdmins, and add this group to sudoers like:

```
# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL
%WARTHOGS\\UbuntuAdmins ALL=(ALL) ALL
```

Configure default domain, HOME directory and default shell

With PowerBroker Identity Service, it is possible to customise the user's desktop environment in multiple ways. You can configure the HOME directory, login shell, domain prefix, etc.

This is accomplished with the command /opt/pbis/bin/config.

/opt/pbis/bin/config --list will give you a list of all the supported settings while /opt/pbis/bin/config --show SETTING displays the current value for a given SETTING.

To set the domain prefix run:

```
~$ sudo /opt/pbis/bin/config UserDomainPrefix [Domain]
```

Set AssumeDefaultDomain to 'true' to avoid entering domain names constantly:

```
~$ sudo /opt/pbis/bin/config AssumeDefaultDomain True
```

Change the default shell from the default /bin/sh to /bin/bash for example:

```
~$ sudo /opt/pbis/bin/config LoginShellTemplate /bin/bash
```

The default HOME directory is set to /home/local/<DOMAIN>/<USERNAME>. It can be changed with:

```
~$ sudo /opt/pbis/bin/config HomeDirTemplate %H/%D/%U
```

Where:
- %H = /home
- %D = Domain Name
- %U = Username

And finally set a specific Active Directory security groups:

```
~$ sudo /opt/pbis/bin/config RequireMembershipOf "[Domain]\\
[SecurityGroup]"
```

At this point we have working logins via GDM, the console or a remote shell like ssh, bash is set as the default shell and the home directory structure matches a standard user directories.

Leaving a domain

To remove your Ubuntu Desktop machine from the domain simply run the following command:

```
~$ sudo domainjoin-cli leave
Leaving AD Domain:   WARTHOGS.BIZ
SUCCESS
```

You can then confirm that the machine really left the domain with:

```
~$ sudo domainjoin-cli query
Name = ad-desktop-1
Domain =
```

The domain is empty.

Alternatives to PowerBroker Identity Service Open

If, for some reason, Powerbroker Identity Service Open is not suitable for your environment, there are other options that can help you integrate Ubuntu into Active Directory for authentication and identity management.

PowerBroker Identity Service Enterprise

BeyondTrust, the company behind Powerbroker Identity Service Open, also offers an enterprise version of its interoperability software. Powerbroker Identity Service Enterprise offers a range of features useful in a large-scale deployment and a group policy for use with supported platforms such as Ubuntu. It also includes an ADC management console, integration with the Active Directory Users and Computers MMC snap-in, extended auditing and reporting features, an NIS migration assistant, SSO capabilities for Apache and Samba, and more.

NSS and PAM configured for LDAP and Kerberos

Microsoft Active Directory is based, in part, on the LDAP and Kerberos standards. Both of these protocols are well supported in the Linux world. LDAP can be used as a database for NSS, and both LDAP and Kerberos can be used as an authentication backend with PAM. Combined together, you can get the same result as using Powerbroker Identity Service Open, without the disconnected operations.

Unix user and group accounts requires a certain number of attributes that are not present by default in Active Directory. Starting with Windows Server 2003 R2, the Active Directory schema have been extended to include attributes conforming to RFC 2307 - 'An Approach for Using LDAP as a Network Information Service', which defines the LDAP attributes required by Unix and Unix-like system, such as Ubuntu. A role service, Identity Management for UNIX, is available in Active Directory to extend functionalities precisely for that purpose. From there on, Active Directory can be used as a store of Unix users and groups as is, without the help of third-party tools. This approach may prove more flexible than using Powerbroker Identity Service Open, at the cost of being more management heavy.

On the Ubuntu side, a set of configurations for the components involved will need to be documented and deployed. Starting with release 7.10, Ubuntu ships with a template-based framework for configuring NSS and PAM called auth-client-config, which could be used to ease the process.

On the Active Directory side, RFC2307 attributes of users and groups would need to be set and managed. While installing the Identity Management for UNIX role service extend the Active Directory Users and Computers MMC snap-in to expose these attributes and allows for setting their value manually, a large deployment will certainly need some sort of tool to automate the process. This needs to be taken into consideration.

Plain winbind

As explained in the 'Overview of Powerbroker Identity Service Open' section, Powerbroker Identity Service Open is based on winbind, a part of the Samba open source project. Winbind itself includes PAM and NSS modules to authenticate and resolve users and groups to an Active Directory. It is generally used alongside Samba for file sharing services, where it exposes Windows domain users as local Unix users.

Its use, however, is not restricted to Samba: it would also work well with any services that use NSS and PAM for user management. For example, our demonstration involving Dovecot would work just as well with a properly configured winbind installation. This would involve modifying /etc/nsswitch.conf (the system NSS configuration file) and /etc/pam.d/ (the PAM system configuration directory) to use winbind.

One of winbind's tasks is to keep a mapping of Unix numerical user and group ID for Windows user and groups. By default, winbind assigns Unix uid and gid sequentially as users and groups are being looked up. The result is, obviously, rather random and will vary from one machine running winbind to another. This will pose a problem if your infrastructure requires Unix and Unix-like machines to share a coherent uid and gid namespace; that would be the case if, for example, you were to share files using the Unix-native NFS protocol. Fortunately, winbind can be configured to use a so-called idmap backend that can be shared among multiple winbind instances. The job of these idmap backends is to store the Windows SID to Unix id mapping, ensuring that users and groups ID are consistent across all machines using the same backend. For example, such an idmap could ensure that the Active Directory user WARTHOGS\bob has a user ID 13897 on both server ubuntusrv1 and ubuntusrv2, making file permissions manageable and consistent when files are being shared between the two. Various idmap backend are available, and winbind can also be configured to derive Unix ID algorithmically based on the Windows RID, just like Powerbroker Identity Service Open.

If you wish to learn more about winbind, the best reference remains the Official Samba How to and reference guide

SSSD and Active Directory

sssd can be used to authenticate user logins against an Active Directory using sssd's 'AD' provider. In previous versions of sssd, it was possible to authenticate using the 'LDAP' provider. However, when authenticating against a Microsoft Windows AD Domain Controller, it was generally necessary to install the POSIX AD extensions on the Domain Controller. The "AD" provider simplifies the configuration and requires no modifications to the AD structure.

Conclusion

Ubuntu Desktop is an increasingly common requirement for software development teams. IT departments want to be able to support and enable the development teams who are asking for Ubuntu and so integration in to the existing directory solution is essential. This document has explored methods of integration for reasons of IT policy compliance, ease of administration and end-user experience. We have looked at various ways to enable user accounts configured in AD environment to seamlessly log on to Ubuntu workstations.

There are both free and paid-for options to make the integration of Active Directory straightforward and manageable. Depending on your IT policy requirements, the time you have to commit to a solution and the expertise of your team your business should be able to achieve a good level of integration.

Further reading:

PowerBroker Identity Service Open Source documentation
The reference documentation for PowerBroker Identity Service both Open and Enterprise version

The Ubuntu Server Guide
A good starting point for everything Ubuntu-related, including sections on LDAP, Kerberos and Samba

IETF RFC 2307 - An Approach for Using LDAP as a Network Information Service
For directory administrator interested in knowing the exact purpose of all LDAP attributes used by NSS

Kerberos Explained
A dated but excellent article on Microsoft's implementation of the Kerberos protocol. The explanations are platform-agnostic

© Canonical Limited 2019. Ubuntu, Kubuntu, Canonical and their associated logos are the registered trademarks of Canonical Ltd. All other trademarks are the properties of their respective owners. Any information referred to in this document may change without notice and Canonical will not be held responsible for any such changes.

Canonical Limited, Registered in England and Wales, Company number 110334C Registered Office: 12-14 Finch Road, Douglas, Isle of Man, IM99 1TT VAT Registration: GB 003 2322 47

ubuntu
Delivered by Canonical