KETS Active Directory Operations Guide

Kentucky Department of Education

Version 1.0.3 *11/18/2014*

Contents

Table of Contents

Table of Contents

Change Log

Version	Date	Editor	Description
2/19/14	John Fabry	v0.1	Document Creation
4/4/14	John Fabry	v0.2	First Draft Completed
4/22/14	John Fabry	v0.5	First Round Of Revisions Completed
4/24/14	John Fabry	V0.9	Additional proofreading and AD structure graphic added
4/29/14	Garrett Dutton	V0.9.1	Removed section about DHCP conflict detection and added section on Dynamic DNS config in DHCP. Reformatted doc to be in line with KETS Operations Guides.
5/13/14	John Fabry	V1.0	Final MADS proofreading.
5/23/14	John Fabry	V1.0.1	Updated proofreading from Kenny Brakefield.
7/7/14	John Fabry	V1.0.2	Added sections on applications connecting to AD and tertiary DNS entries for clients.
11/14/14	John Fabry	V1.0.3	Added information on forest-wide machine name uniqueness and AD Recycle Bin.

Check this space throughout the document for important information or links to additional content and documentation.

1. Introduction

Welcome to the Microsoft Windows Server 2012 R2 Active Directory Operations Guide. This guide outlines the technologies and steps involved in administering the Kentucky Education Technology System (KETS) Active Directory Domain Services 2012 R2 environment. The focus of this guide is to convey the necessary tasks for carrying out routine operations required to administer your district's Active Directory 2012 R2 system. Specifically, this guide will provide explicit guidance on KETS-specific aspects of Active Directory while pointing to authoritative sources from Microsoft for general information related to Active Directory Domain Services.

The KETS deployment of Microsoft's Windows Server 2012 R2 Active Directory Domain Services (AD DS) is not complex; it's simply large. Many of the changes when moving from the previous Windows Server 2008 environment to the new Windows Server 2012 R2 one are optional; while you are encouraged to leverage the new tools and capabilities of Windows Server 2012 R2 AD DS, the old tools you may be familiar with will continue to function. For instance, while you can manage users through the new Active Directory Admin Center (ADAC) and Windows PowerShell now, you can still continue to use Active Directory Users and Computers (ADUC.)

This guide will also include brief discussions regarding several other services that run on the domain controllers in conjunction with Active Directory Domain Services and that are commonly integrated with it, such as DNS and DHCP.

1.1. Audience

This guide was written and is kept up-to-date for the technical administrators and user managers of Kentucky school districts' directory services system.

1.2. Technologies/Terminologies

There are acronyms and technology terms that are used when discussing Active Directory Domain Services. The technology field is riddled with these terminologies which can cause confusion if not properly defined. It's important, though, to again reiterate that technical administrators in this K-12 environment are the audience for this document.

The term "***KETS***" will be utilized throughout this document, referring to all users and technologies which utilize enterprise services delivered to the 175 (including KSB and KSD) school districts by ***KIDS*** (the Office of Knowledge, Information, and Data Services) which was formerly the Office of Education Technology (OET.) KIDS is the technology office of the Kentucky Department of Education (KDE.) ***Messaging and Directory Services*** or ***MADS*** is the team in the Office of KIDS responsible for maintaining the AD DS environment along with Office 365.

Windows Server 2012 R2 Active Directory Domain Services (also known as Active Directory or ***AD DS***) is utilized by KETS. The Active Directory Domain Controllers, the servers which run the service, utilize Windows Server 2012 R2 Hyper-V technology. Microsoft's Windows Server 2012 R2 ***Hyper-V*** provides virtualization of operating systems and their services. KETS also utilizes Domain Name Services (***DNS***) which resides within Active Directory as well as external to Active Directory on other platforms.

Office 365 or ***O365*** is Microsoft's cloud service which provides access to email, instant messaging, web conferencing, and collaboration. While not directly a part of AD DS, O365 is connected to AD through the Online Provisioning System (***OLPS***.) OLPS allows for accounts created in AD to automatically have a corresponding account created in Office 365.

Password Change Notification Service or ***PCNS*** allows for passwords in Active Directory and Office 365 to be in sync. Since Microsoft Organization IDs (***OrgIDs***) will be the same as each user's SMTP address and that address is used as the User Principal Name (***UPN***) in Active Directory, users can have the same login between Active Directory and Office 365. PCNS runs

as a service on every domain controller in the environment in order to push password changes from AD DS to O365.

Remote Server Administration Tools or ***RSAT*** is a package created by Microsoft to allow client workstations to install the same applications which would be present on server machines running the services in question. For example, installing the RSAT package will allow you to manage AD DS from a remote client rather than requiring anyone to log in to a domain controller locally or via RDP.

Active Directory Users and Computers or ***ADUC*** is a Microsoft Management Console (***MMC***) snap-in that allows AD DS administrators to manage security principals in Active Directory. This utility was available in Windows Server 2008 and continues to function with AD in Windows Server 2012 R2.

Active Directory Administrative Center or ***ADAC*** is a new tool provided by Microsoft to streamline the management of security principals in AD DS. This was first introduced in Windows Server 2008 R2 and is one of the primary means by which to manage AD DS in Server 2012 R2. It provides a GUI front-end and actually runs PowerShell commands on the back-end. It is capable of giving a transcript of the PowerShell commands it utilizes.

PowerShell is a command line-based scripting language that can be used to administer objects in AD DS. While many of the tasks that can be accomplished through PowerShell can be accomplished via other utilities as well (eg. ADUC and ADAC), PowerShell provides a means by which to programmatically make changes. This can be extremely useful to help AD DS administrators save time with bulk modifications to the directory.

1.4. Document Feedback

If you have ideas for improving this document, such as adding additional information or clarifying existing content, please send them to your KETS Engineer so they can be considered for future versions.

1.5. Document Updates/Location

This document will be updated and enhanced over time. Please check for new versions periodically at http://education.ky.gov/districts/tech/Pages/Administration-and-Install-Guides.aspx

2. "Need to Know" Items

This section will outline the items that are not necessarily "task"-oriented but are extremely important to the administration of AD DS. Many of these discussions are expanded upon in Section 3 where pertinent. Please read and understand all of the items that follow as they will serve as the foundation of the Active Directory system.

There are many different Active Directory Domain Services management tools available for administrators. Some are provided by Microsoft while others may be available from third parties. Some common tools are:

- Active Directory Users and Computers (ADUC)
- Active Directory Administrative Center (ADAC)
- CSVDE, LDIFDE, etc.
- DSADD, DSMOD, etc.
- Active Directory Windows PowerShell module
- Other 3rd party tools (AD Infinitum, etc.)

3rd party tools are not supported by KIDS and should be used only at the district's discretion.

Using any of these tools will require authentication against Active Directory. The level of permissions required will depend on the task being attempted. For example, just querying user objects in the Staff OU of a domain can be done by any authenticated user in any domain in the forest. Actually making changes to those users or adding new ones, however, would require the use of an account which has membership in a privileged security group, such as DIST Support Admins or DIST Staff User Admins.

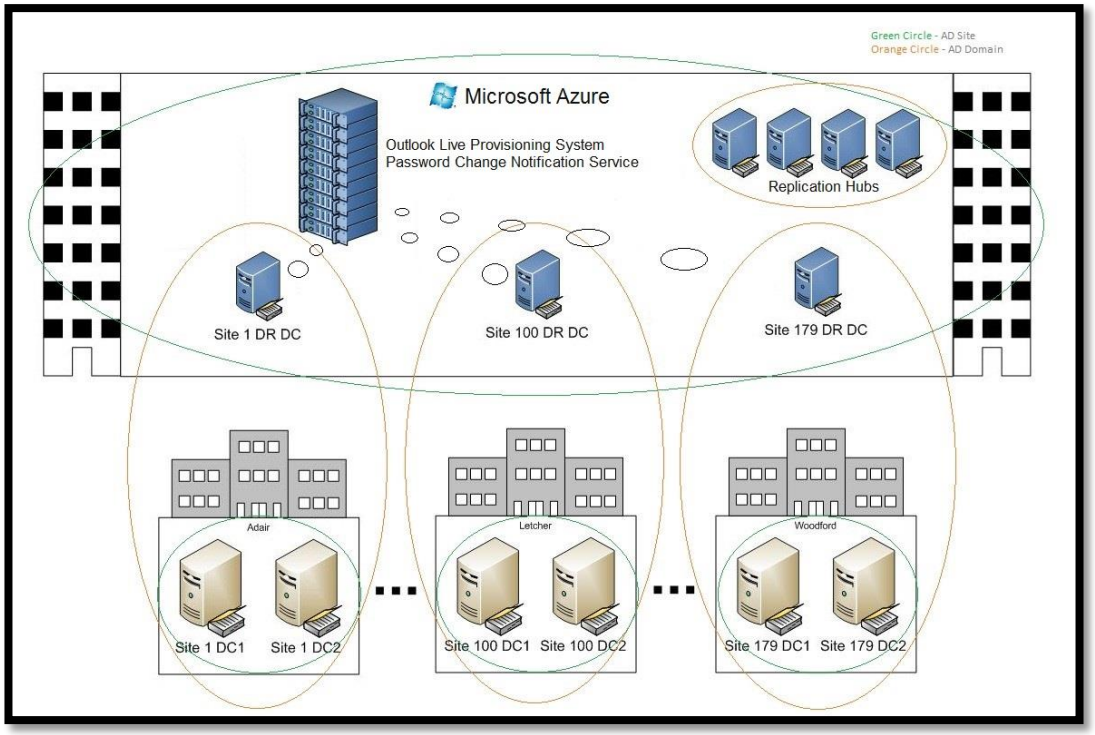
2.1. Technologies

2.1.1. Active Directory

The KETS Active Directory (AD) environment is built on Windows Server 2012 R2 Domain Services. Active Directory is responsible for user authentication and authorization

throughout many services within the district environment. Users rely on DNS within AD as well as 'external' DNS when required. AD is also integrated with DHCP, WINS, and NTP.

The design of Active Directory for KETS exists as a classic hub-and-spoke topology, consisting of an 'empty' root domain (KETSDS.NET) with 180 sub-domains. AD replication is linear from each district domain to the hub-site, replicating on a one-hour interval. Each district domain has two AD domain controllers which reside physically within the district on KIDS managed hardware. Each district also has an additional AD domain controller which resides in Microsoft's Windows Azure Infrastructure-as-a-Service (IaaS) cloud. These tertiary domain controllers exist in an environment that is called 'DR AD' (Disaster Recovery Active Directory.) Backups of the environment occur against the tertiary, IaaS DCs to ensure the best availability.



The basics of AD DS under Windows Server 2012 R2 are described more fully here: http://technet.microsoft.com/en-us/library/hh831477.aspx

2.1.2. DNS

The KETS DNS (Domain Name Service) environment is also built on Windows Server 2012 R2. DNS allows for the resolution of a human-readable hostname to a computer-readable IP address. A district's domain controllers are authoritative for the ***district***.*ketsds.net* internal DNS zone. This zone is tightly linked with AD DS as it is an Active Directory Integrated zone. Information on that zone is replicated to both of the district's local domain controllers, as well as the tertiary domain controller in Windows Azure. Similarly, there is another AD Integrated reverse lookup zone for the 10.x.x.x network. This contains reverse or PTR (pointer) records that allow users and applications to query DNS with an IP address and receive a hostname back.

Along with these AD Integrated zones, there are also secondary zones for the ***district.kyschools.us*** zone each district has. As a secondary zone, the domain controllers are not authoritative for the zone; they are just hosting a copy of it that they periodically update from the authoritative server to help speed up name resolution within the district.

Note that district staff are not delegated permission to DNS. Any necessary changes should be accomplished by opening a ticket with the KETS Service Desk.

2.1.3. DHCP

DHCP (Dynamic Host Control Protocol) allows for clients to automatically receive an IP address based upon location and network setup. The KETS network privateaddress space is allocated out of the 10.x.y.z address range. This private address range is not Internet-routable, but is routable intra-district and inter-district. Typically, each district is given one class B network, meaning that in the previous example the district has one number in the **x** octet. In some cases for larger districts, though, multiple class B networks may be provided. So for example, Providence Ind. may be provided the 10.5.y.z network.

That network is further subdivided in to 16 class C networks. The default configuration is the following, where **x** is the district's provided 2nd octet. Each network has a /20 subnet mask

(255.255.240.0 in CIDR notation.) So for instance the first network listed runs from 10.x.16.1 – 10.x.31.254.

- **10.x.16.0**
- **10.x.32.0**
- **10.x.48.0**
- **10.x.64.0**
- **10.x.80.0**
- **10.x.96.0**
- **10.x.112.0**
- **10.x.128.0**
- **10.x.144.0**
- **10.x.160.0**
- **10.x.176.0**
- **10.x.192.0**
- **10.x.208.0**
- **10.x.224.0**

You may notice that this is only 14 class C networks rather than 16. The first network (10.x.0.0) and the last network (10.x.240.0) are not created by default. In the instance that districts need a network with a small number of IPs – for small elementary schools, bus garages, etc. – the space for those two class C networks will be further divided into even smaller subnets in order to allow districts to make more effective use of their allotment of addresses. If necessary, they can also be created as a /20 network.

Along with providing IP addresses, DHCP also allows for the setting of DHCP options. These options can be set at both the subnet level and the server level, with server options being applied to every scope. It is possible to create a server option and then set a scope option which will override it for just a particular scope. DHCP options can handle tasks like providing DNS servers for clients to use, directing new clients to a particular server for PXE boot, and more. Some vendors also have created their own custom options.

Districts have the ability to view DHCP settings via the DHCP MMC snap-in, but they are not delegated ability to change the settings. Any change requests should be submitted via a ticket with the KETS Service Desk.

2.1.4. Group Policy

Group Policy is a feature of the Microsoft Windows NT family of operating systems. It provides for the centralized management and configuration of computers and remote users in an AD DS environment. Essentially, it provides a means by which to control what users can and cannot do on a computer network. Although Group Policy is most commonly used in enterprise environments, its usage is also common in schools and small businesses to restrict certain actions which may pose potential security risks: e.g. blocking the Windows Task Manager, restricting access to certain folders, disabling downloaded executable files, etc.

Most commonly, GPOs (Group Policy Objects) are applied to OUs (Organizational Units) in Active Directory. This allows administrators to easily create different policies for teachers and students at different locations depending on how the directory service is structured.

2.1.5. WINS

WINS, or the Windows Internet Name Service, is a legacy networking protocol that is Microsoft's version of NBNS (NetBIOS Name Service.) WINS provides a central repository which maps a network IP address to a NetBIOS name. Unlike DNS, NetBIOS utilizes just the host name; there is no domain information. The name is also limited to a maximum of 15 characters.

Also unlike DNS, WINS also helps to provide a mapping to which services are available where in the network. For example, a client may register itself as both a workstation and as a file server if it has network shares set up on it. While WINS is typically eschewed by modern applications in favor of DNS, it's still occasionally needed by legacy systems. Even modern clients will still register themselves with WINS.

2.1.6. NTP

NTP, or Network Time Protocol, allows for many machines across a dispersed network to keep their clocks in sync with a relatively low tolerance. Time synchronization is very

important in AD DS as machines with a skewed time may be unable to successfully authenticate against their target domain controllers, even with the proper credentials. Domain controllers in Active Directory are automatically configured to act as time servers via the Windows Time Service. Likewise, domain joined clients are automatically configured to sync time with their nearest domain controller.

2.2. Support

For the most part, the KETS implementation of Windows Server 2012 R2 AD DS is not heavily customized. Almost all documentation from Microsoft is valid for our environment. However, if you are not sure that vendor documentation is correct from Microsoft or otherwise, please contact the KETS Service Desk for clarification.

For any requests or other work in Active Directory which requires administrative permissions greater than what is delegated to the district DIST Support Admins group, please contact the KETS Service Desk so that the issue can be routed to the Messaging and Directory Services (MADS) team.

It is worth mentioning that you should check with your KETS Engineer (KE) prior to purchasing any software or appliances which require special permission to Active Directory. KIDS will not grant Domain Admin rights to any district accounts which could impact your ability to use some 3rd party tools. If you have any concerns about this, speak with your KE before purchasing items that connect back to Active Directory to make sure they will function with the permission available to you.

2.2.1. KETS Service Desk Support

To contact the KETS Service Desk, the following means are available:

- Phone
  - Local – 502.564.2002
  - Toll Free – 866.538.7435
- Email – kesthelp@education.ky.gov

2.3. Responsibilities

2.3.1. KIDS

KIDS is responsible for most technical maintenance and support of systems in the KETS Active Directory environment.

2.3.2. District

Districts are responsible for the physical environment which houses the Active Directory services, aside from the virtual machines located in Windows Azure. This area should stay physically secure and temperature controlled. The district is also responsible for user/object administration as well as any licensing that is required beyond that of the domain controllers themselves.

2.4. User Principal Name

The User Principal Name (UPN) attribute, which exists on all user objects in Active Directory, was created for companies that wanted a unique user definition to the login process which isn't bound by domain designs. The UPN in the KETS environment is set by OLPS to be equal to the user's primary SMTP address (eg. *foo.bar@providence.kyschools.us.*) All users are encouraged to log in to domain resources using this UPN. This provides several advantages to end users. The advantages include giving the users one less thing to remember as all users utilizing mail will need to know their email address but now will not have to know their *domain\user* credentials on top of that.

Note: District users will continue to have the capability to log in with domain\user if the district or user so desires.

2.5. Password Synchronization between AD and Office 365

Passwords between Active Directory and Office 365 are kept in sync through the OLPS system which leverages Password Change Notification Service (PCNS.) PCNS is configured on all of the KETS domain controllers. When a user's password is changed that password is

synced through OLPS to Office 365. This allows for the same password to be used between AD DS and the Office 365 collaborative suite.

It is important to keep this in mind when using either Group Policy or Fine-Grained Password Policies to define a minimum password length for users in Active Directory. While it is possible, though not recommended, to allow for very short passwords in AD DS, those passwords will not sync properly to Office 365 if they do not meet the minimum requirements as specified for Office 365. The options for settings password requirements in AD are discussed later in this document. For information regarding the minimum password requirements in Office 365, please reference the KETS Office 365 Operations Guide (KOOG), which is available here:

http://education.ky.gov/districts/tech/Pages/Administration-and-Install-Guides.aspx

2.6. Availability

In every district there are two Active Directory domain controllers housed locally, one being a global catalog (GC) which is used primary for Universal Security Group membership lookups at logon. Along with these local domain controllers there is a tertiary domain controller hosted in Microsoft's Windows Azure cloud. This cloud houses a total of 180 tertiary domain controllers, one for each sub-domain in AD DS. The primary purpose of this tertiary domain controller in Windows Azure is to interface with the provisioning system, i.e. OLPS, which makes corresponding accounts in Office 365 when accounts are created in AD DS. They also provide off-site system state backups for each districts' AD environment.

3. Administrative Tasks

3.1. AD Administration

Active Directory Domain Services provides a means to manage individual users, computers, groups, and more. It provides a hierarchy in which users and computers can be more effectively organized. Microsoft provides a comprehensive look at AD DS here:

http://technet.microsoft.com/en-us/windowsserver/dd448614.aspx

KIDS provides a skeleton structure in AD DS to help dictate where objects go. Immediately under each district's domain, e.g. *districtName*.ketsds.net, are OUs for Leadership, Staff, Students, and Workstations. These OUs already have permissions set on them for select, pre-created groups within each domain to help districts keep administration granular. This is discussed in detail later in the document. Within these OUs, however, the structure is up to the discretion of each individual district.

Management of Active Directory in the districts will be handled via remote connections through tools either provided by Microsoft or by 3rd parties. This document will only discuss Microsoft's tools, though plenty of other tools from 3rd parties exist. It is worth discussing the use of any 3rd party utilities with your KETS Engineer, though, prior to procuring them as any tools requiring special permission to the domain may not work in our environment.

To acquire Microsoft's tools for any Windows client machines you can download the Remote Server Administration Tools (RSAT) package from Microsoft. These packages are based on your client operating system, and the most common ones are available below:

RSAT for Windows 8.1 - http://www.microsoft.com/en-us/download/details.aspx?id=39296

RSAT for Windows 8 - http://www.microsoft.com/en-us/download/details.aspx?id=28972

RSAT for Windows 7 - http://www.microsoft.com/en-us/download/details.aspx?id=7887

Administrative Tasks

Administration of objects in Active Directory, especially user and group objects, can rarely be considered standalone in the KETS environment. These objects also tie into Office 365. While many of the details of that interaction can be found in the KETS Office 365 Operations Guide (KOOG), which can be found here, it will also be mentioned in this document where relevant.

One of the biggest changes in Server 2012 R2 Active Directory is that administrators have many more options as far as tools are concerned for managing users. In the previous Server 2008 environment, administrators typically used:

- **Active Directory Users and Computers (ADUC)**
- **ADSIEdit**
- **LDP**
- **Ldifde, Csvde**
- **DSTools (dsmod, dsadd, etc.)**

These tools are all created by Microsoft and are included in the Remote Server Administration Tools (RSAT) package. Along with these, some district admins also chose to use 3rd party tools such as AD Infinitum, custom scripts, etc. While it is anticipated that all of these 3rd party tools will continue to work in the new Server 2012 R2 environment, KETS cannot make any guarantees as it will depend on how each individual 3rd party utility has been coded. Any new tools available from Microsoft can supplement or replace any other tools at the discretion of individual administrators. Some of the new tools are:

Note that both of these new tools are included in the RSAT package.

- **Active Directory Administrative Center**
- **Windows PowerShell**

You can find more details regarding the use of the Active Directory Administrative Center (ADAC) here:

http://technet.microsoft.com/en-us/library/dd560651(v=WS.10).aspx

Microsoft has details on all of the available PowerShell cmdlets for AD DS here:

http://technet.microsoft.com/en-us/library/ee617195.aspx

3.1.1. Administrative Access

Rights to perform AD DS administration in each district is handled by AD group membership. The following lists the various groups in Active Directory and their respective rights. Note that each of these groups can be found in the **_District Admins > Users and Groups** OU.

- **DIST Support Admins**
  - User management access for Staff and Student accounts
    - Create/delete users
    - Reset passwords
  - Management access through various AD DS tools and KETS Control Panel
- **DIST Staff User Admins**
  - User management access Staff only
    - Create/delete users
    - Reset passwords
  - Management access through various AD DS tools and KETS Control Panel
- **DIST Student User Admins**
  - User management access for Students only
    - Create/delete users
    - Reset passwords
  - Management access through various AD DS tools and KETS Control Panel

3.1.2. AD Basics

The goal of this document is to provide a comprehensive overview of AD DS-specifics *as they relate to KETS*. Microsoft provides a very large amount of documentation regarding the general workings of Active Directory which are the same in every AD DS environment. For a generic overview of AD DS, Microsoft maintains the following article:

http://technet.microsoft.com/library/hh831484

The next provided article also highlights AD DS-specific features which are new to Windows Server 2012 and 2012 R2. Note that not all of the listed features will be relevant to

district staff, as many focus on the high-level administration of AD DS; the link is mainly provided for the edification of anyone who may be curious:

http://technet.microsoft.com/library/hh831477.aspx

3.1.3. OU Structure

As was mentioned previously, the domain for each district has a standardized OU configuration. Districts have administrative privileges over many of the OUs, and the sub-OU structure is open to the district's dictation. This section will cover some of the more relevant top-level OUs and the corresponding availability to each district.

Some of the objects described below are actually containers rather than OUs, giving them slightly different properties. Containers are included in the directory by default. Microsoft provides authoritative documentation on them here: *Administration of Default Containers and OUs*

- **_District Admins**
  - A KETS-maintained OU which houses service accounts and groups that are created and delegated by MADS but which are designed for use by the district. For example, members of the DIST Support Admins group can modify the membership of the groups within this OU.
- **_Enterprise Admins**
  - Contains KETS-maintained users and groups to which the district does not have access.
- **Builtin**
  - Contains a collection of Security Groups which are automatically created in every AD DS environment. District staff have no access to this container.
- **Computers**
  - A default container in every AD DS environment, any new computer objects joined to a domain *without* being pre-created in AD will be placed here by default. It is important to note that, being a container, Group Policy does **not** apply here so administrators will want to move objects out of this container and into either the Local Servers OU or the Workstations OU as discussed later.
- **Domain Controllers**
  - Houses the AD DS object for each domain controller in a given domain. District staff have no access to this OU.
- **Enterprise Servers**
  - Top-level OU that is home to the computer objects for non-AD DS enterprise services provided by KETS (e.g. ePO, WSUS.) District staff have no access to this OU.

- **Leadership**
  - In the previous on-premises Exchange environment, this OU was designed for staff accounts which had a larger mailbox than those in the Staff OU. Today it has no special purpose. While accounts created in this OU will still be usable and will provision to Office 365, there is no functional difference between them and those in the Staff OU. Accounts placed here will automatically provision a mailbox with a **@*district*.kyschools.us** address. The pre-created _Exchange Resources sub-OU is designed to hold Distribution Groups for Leadership members. It is also important to note that items placed in the _Groups sub-OU will **not** provision to Office 365.
- **Local Servers**
  - Servers as a place where districts can place their own servers. A sub-OU structure, if any, is up to the district.
- **Staff**
  - Designed to contain all staff and faculty members in a district. While many districts choose to create a sub-OU structure based on schools, it is ultimately a district decision. Accounts created in this OU will automatically provision an Office 365 account with a **@*district*.kyschools.us** address. The pre-created _Exchange Resources sub-OU is designed for Distribution Groups for Staff members. It is also important to note that items placed in the _Groups sub-OU will **not** provision to Office 365.
- **Students**
  - Holds all student accounts in a domain. While many districts choose to make sub-OUs based on school location or expected graduation year, the structure is ultimately a district decision. Accounts created in Students will automatically provision accounts to Office 365 with a **@stu.*district*.kyschools.us** address. The pre-created _Exchange Resources sub-OU is designed for Distribution Groups for Student members. It is important to note that items placed in the _Groups sub-OU will not provision to Office 365.
- **System**
  - A container housing various sub-containers for additional features of AD DS, such as Password Settings Objects (PSOs) and SCOM configurations. District staff have no access to this container. Modifications to it may be necessary for SCOM implementations and PSO requests; in those instances a ticket should be created with the KETS Service Desk.
- **Users**

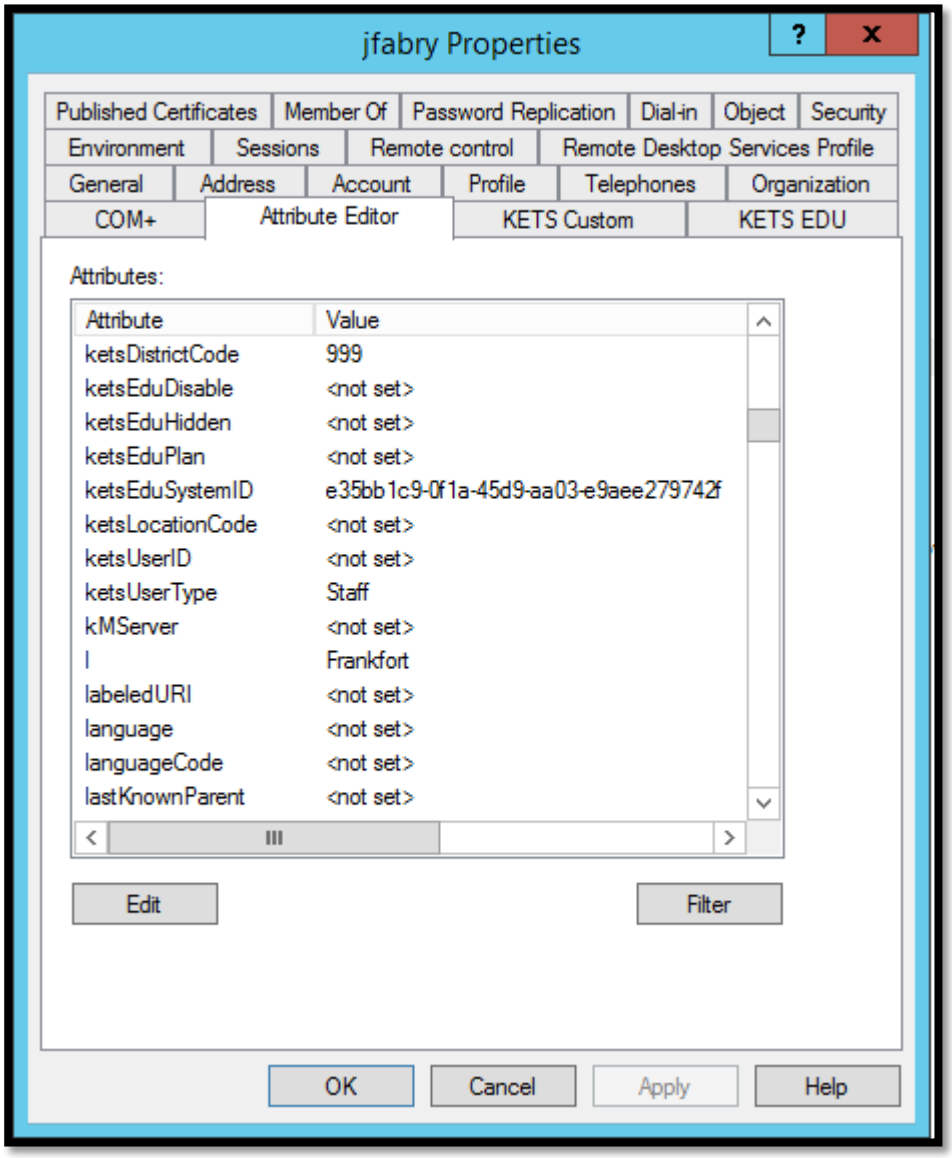
    - A built-in, generic container for user objects in AD DS. Given that there is no option to provision accounts to Office 365 from this container, its use is not recommended.
- **Workstations**
    - Designed to house computer objects for all of the client workstations in a district. Many districts opt to create sub-OUs per location where a workstation could exist, though it is ultimately a district decision what sub-structure to use, if any.

3.1.4. AD Attributes

AD DS objects contain attributes. Attributes are various topics which may or may not contain data about a particular object. For example, the **mail** attribute will provide a user object's SMTP address. The attributes which exist in a given AD DS environment will vary depending on how the AD DS Schema has been configured. In the KETS AD DS forest, the schema has been extended only very selectively for enterprise projects to ensure the maximum stability and uniformity through the environment while minimizing bloat and complexity. With one exception, the Schema has only been extended with first-party, Microsoft-generated attributes which were necessary for things like Microsoft Exchange. A full list of attributes offered by Microsoft is given here:

http://msdn.microsoft.com/en-us/library/windows/desktop/ms675090(v=vs.85).aspx

However, KETS has done one Schema extension to add attributes generated by MCS (Microsoft Consulting Services) in order to allow OLPS to connect AD DS to Office 365. These attributes all begin with **kets** and are easily visible via the Attribute Editor tab in ADUC. If you do not see the Attribute Editor tab, you may need to go to the **View** menu and select **Advanced Features**. They are also available on the custom **KETS EDU** tab which can be used with ADUC. Using this custom tab requires the installation of a unique .msi file created for KETS. You can get this installer by contacting the KETS Service Desk.



The following two of these attributes CAN NOT be populated by the district. OLPS is authoritative for them, and attempting to modify these values manually can result in problems for an accounts link to Office 365.

It is important to understand these custom attributes and be familiar with their uses. Administrators can leverage them for custom programming/scripting, LDAP queries, PowerShell cmdlets, and more.

- **ketsDistrictCode**
  - Purpose: Value relates to the three-digit district number.

- AD Flow: Flows to *extensionAttribute14* in AD DS.
- O365 Flow: Flows to *CustomAttribute1* in O365.
- Used to set the '*Company*' attribute in Office 365 – the district name (e.g. 'providence') is set in the Company field for mailboxes.

- **ketsEduSystemID**
  - Purpose: Immutable ID created by OLPS for uniqueness to track and manage each object it handles.
  - O365 Flow: Flows to *Name* in O365.

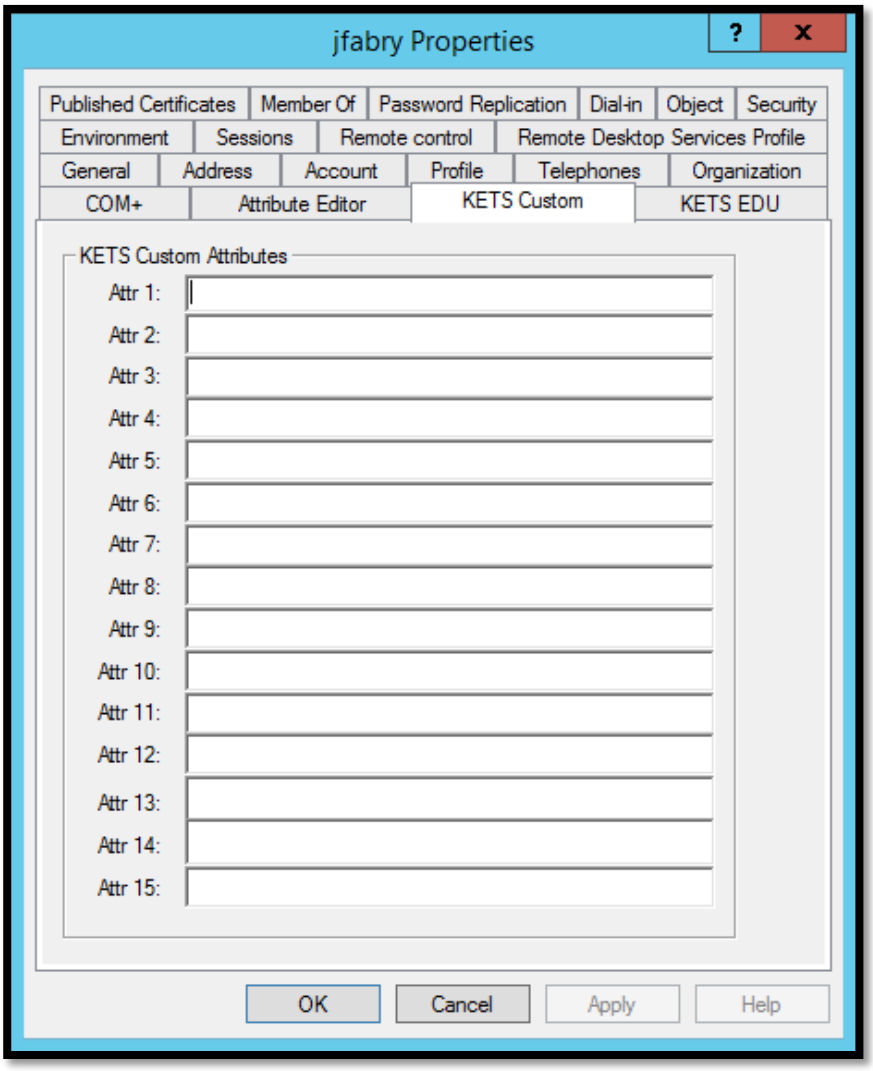
The remaining attributes, however, are populated either directly or indirectly by the district staff. This may happen by OU placement, choices on the **KETS EDU** tab, or through mass modification through PowerShell, LDIFDE, etc.

- **ketsEduDisable**
  - Purpose: Disables access to all Office 365 cloud services.
    - Note: The mailbox will continue to receive mail and the OneDrive For Business content will be preserved.
  - Values: TRUE or FALSE
  - Default: FALSE
  - O365 Flow: Flows to *BlockCredential* in Azure AD.
- **ketsEduHidden**
  - Purpose: Hides visibility in the GAL (Global Address List)
  - Values**:** TRUE or FALSE
  - Default: FALSE
  - O365 Flow: Flows to *HiddenFromAddressListsEnabled*
- **ketsEduPlan**
  - Purpose: Determines if OLPS will create a mailbox for the specified user.
  - Values: <default> or NoMail
  - Default: <default> *which results in mailbox creation.*
- **ketsUserType**
  - Purpose: To clarify user type.
  - Values: STAFF, STUDENT, or RESOURCE
    - *Note objects set to Resource are not provisioned by OLPS.*
  - Default: <not set>
  - AD Flow: Flows to *extensionAttribute15* in AD.
- **ketsLocationCode**
  - Purpose: Location identifier
  - Value: Entered by district

- Default: <not set>
- O365 Flow: Flows to *CustomAttribute2* in O365

- **ketsUserID**
  - Purpose: System attribute
    - *Do not assign*

Along with these are 15 extra attributes named **ketsCustom1** through **ketsCustom15** which also have their own tab in ADUC if you install the aforementioned .msi file:



Be cautious about putting any sensitive information in these attributes since they can be queried by any other authenticated user in the environment!

These attributes exist for the purpose of district population, and districts may utilize them for any reason they need. They do **not** flow to Office 365.

3.1.5. Password Policies

There are multiple ways to specify a password policy for users in each domain. This can be accomplished via a domain's Default Domain Policy, which is a Group Policy Object that will impact every user in the domain. It can also be done through Fine-Grained Password Policies (FGPPs) that apply only to groups of users within AD DS. FGPPs allow administrators to very granularly set different, highly specific password policies for different sets of users as necessary. It is also worth mentioning that both the password settings in the Default Domain Policy and Fine-Grained Password Policies can be used in conjunction with one another; in those instances users with an FGPP applied will have that take precedence over the Default Domain Policy.

While administrators have the option of making the password policies in AD DS as strong or weak as they may like, there are additional considerations to keep in mind if those AD accounts will also be getting Office 365 accounts. PCNS will sync passwords between AD DS and O365, but that functionality will only work if the AD DS password meets the O365 password requirements. The O365 requirements are outlined in the KETS Office 365 Operations Guide, which is available here: *Administration and Install Guides*

Along with passwords, both the Default Domain Policy and FGPPs give options for managing account lockout settings as well. This empowers administrators to control how many incorrect password attempts are allowed, for example, before an AD DS account becomes inaccessible and for how long that lockout period lasts.

3.1.5.1. Default Domain Policy

The following is an example of a Default Domain Policy:

Account Policies/Password Policy	
Policy	**Setting**
Enforce password history	24 passwords remembered
Maximum password age	30 days
Minimum password age	0 days
Minimum password length	8 characters
Password must meet complexity requirements	Enabled
Store passwords using reversible encryption	Disabled
Account Policies/Account Lockout Policy	
Policy	**Setting**
Account lockout duration	5 minutes
Account lockout threshold	40 invalid logon attempts
Reset account lockout counter after	5 minutes

Storing passwords with reversible encryption allows for passwords to be easily decrypted to plaintext. This is highly insecure and should never be turned on by default for all accounts via policy. Microsoft provides additional details here: *Store passwords using reversible encryption*

The description of exactly what each of these options for the Password Policy mean are outlined here:

http://technet.microsoft.com/en-us/library/cc875814.aspx

Along with the Password Policy, the Default Domain Policy can also specify an Account Lockout Policy that controls how frequently accounts become set into an unusable, locked state based on time duration and the number of incorrect password attempts. Microsoft describes the available lockout settings here:

Note: Locking out an account in AD DS will not lock out the corresponding O365 account. Similarly, O365 accounts cannot lock out AD accounts.

http://technet.microsoft.com/en-us/library/cc757692(v=WS.10).aspx#w2k3tr_sepol_accou_set_tdtx

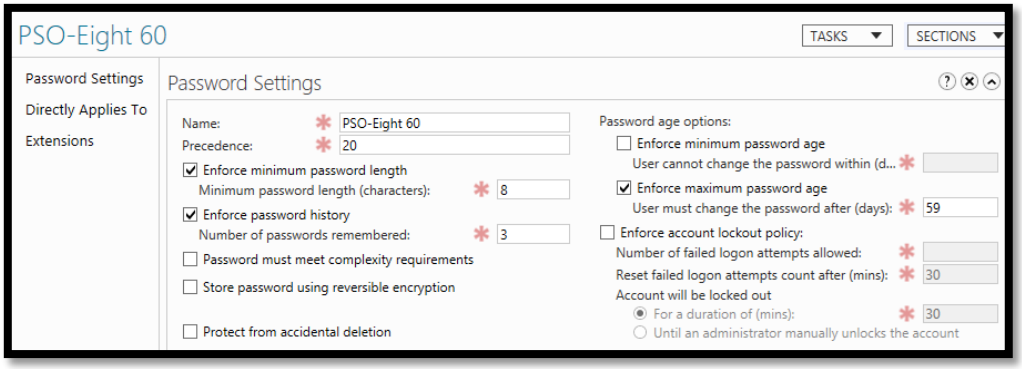
While district administrators have the ability to view the Default Domain Policy, they cannot make modifications to it. To request any changes, contact the KETS Service Desk so a ticket with the requested settings can be sent to the MADS team.

3.1.5.2. Fine-Grained Password Policies

Fine-Grained password policies are comprised of two parts: a Password Settings Object (PSO) and an AD DS group. Each PSO is stored in the System container in each domain. The PSO houses the information related to the policy (e.g. if complexity is required, the minimum length, etc.) Just like the Default Domain Policy, FGPPs contain exactly the same

settings for password and lockout policies. See the links in the previous section for a detailed description.

Below is an example of the data contained in a PSO:



One of the attributes associated with a PSO is called *ms-DSPSOAppliesTo*. This attribute contains the location of the group which the PSO will impact. Any users added as members of that group will begin using the password policy specified by the FGPP rather than the Default Domain Policy.

One attribute on the PSO visible on the screenshot above is Precedence. This number is used to determine which policy applies in the instance that an individual user has more than one FGPP applied to his or her account. This number is automatically determined by the MADS team and will result in the stricter password policy being applied. You do not need to give a Precedence value when opening a ticket.

By default, KETS provides 8 FGPPs in every domain by default. If you wish to have a custom FGPP created, open a ticket with the KETS Service Desk specify what information should be included in the policy so that MADS can create it.

The names of the policies below correspond to the names of the groups in AD DS to which the policies apply. These groups exist in the **_District Admins** OU, within the **Users and Groups** sub-OU. To see which of these policies will work with Office 365, see the KETS Office 365 Operations Guide (KOOG):

http://education.ky.gov/districts/tech/Pages/Administration-and-Install-Guides.aspx

- **DIST Password Policy – None**
  - This policy requires no minimum password length, no complexity, forces no change, and has a password history of zero.
- **DIST Password Policy – Three Never**

  - This policy requires a minimum three character password length, no complexity, forces no change, and has a password history of three (meaning you cannot reuse the last three passwords.)
- **DIST Password Policy – Six Never**
  - This policy requires a minimum six character password length, no complexity, forces no change, and has a password history of zero.
- **DIST Password Policy – Six Complex 60**
  - This policy requires a minimum six character password length, forces complexity, forces a change at 60 days, and has a password history of five (meaning you cannot reuse the last five passwords.)
- **DIST Password Policy – Seven Never**
  - This policy requires a minimum seven character password length, no complexity, forces no change, and has a password history of zero.
- **DIST Password Policy – Eight 60**
  - This policy requires a minimum eight character password length, no complexity, forces a change at 60 days, and has a password history of three (meaning you cannot reuse the last three passwords.)
- **DIST Password Policy – Eight Complex 120**
  - This policy requires a minimum eight character password length, forces complexity, forces a change at 120 days, and has a password history of three (meaning you cannot reuse the last three passwords.)
- **DIST Password Policy – Eight Complex 30**
  - This policy requires a minimum eight character password, forces complexity, forces a change at 30 days, and has a password history of twelve (meaning you cannot reuse the last twelve passwords.)

3.1.5.3. Forcing Password Changes

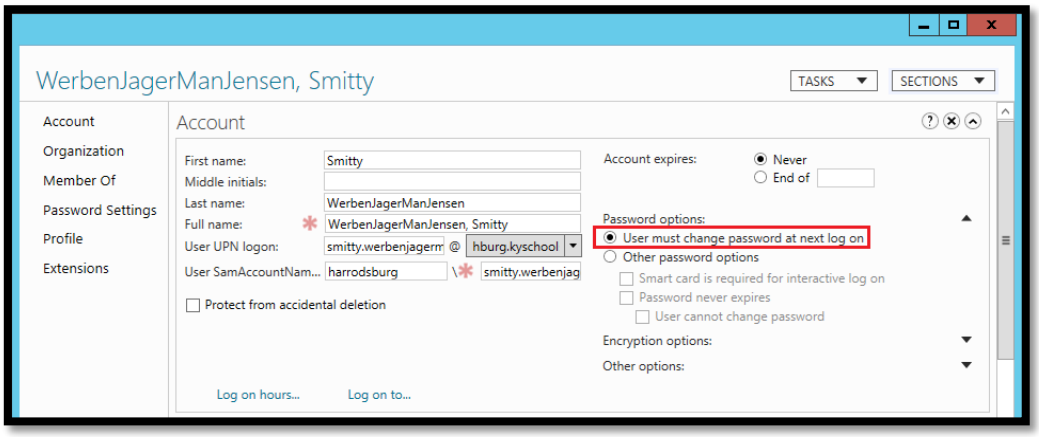
It is important to understand that when adding a user to a group under the purview of an FGPP or when having a modification made to the Default Domain Policy, the impacted user(s) will not be forced to immediately change their passwords to match the new requirements. Instead, the next time the users either opt to or are forced to change their passwords, the passwords must match the new requirements.

If a user had previously never been forced to reset his or her password, then IT administrators have a couple of options for forcing a change. If feasible, users can simply be

instructed to manually initiate a password reset through either a domain joined Windows workstation or the Password Reset page:

https://live.kyschools.us/password.aspx

The other option is to administratively force a password change by having the AD object set to "**User must change password at next logon**". For a small number of users, this can be done through the ADAC GUI, as is shown below:



You can also use PowerShell to accomplish the same. An example of this is:

```
Set-ADUser -ChangePasswordAtLogon:$true -Identity:"CN=WerbenJagerManJensen\,
Smitty,OU=Staff,DC=harrodsburg,DC=ketsds,DC=net" -
Server:"ED242ADGC1.harrodsburg.ketsds.net"
```

The following documentation gives the full explanation of the Set-ADUser PowerShell cmdlet, or you can also run:

```
Get-Help Set-ADUser -detailed
```

http://technet.microsoft.com/en-us/library/ee617215.aspx

For a large number of users, administrators can use Excel to concatenate together a large number of commands. The setting can be made either with Set-ADUser as listed above or via older AD DS tools such as ldifde or csvde. For a tutorial on how to use Excel to concatenate a script together, there is documentation available at the below link in the **Delete Student Mailboxes** folder:

http://bit.ly/madsdocs

3.1.6. Machine Name Uniqueness

3.1.6.1. Background on Unique Computer Names

Unlike Server 2008, Windows Server 2012 R2 AD DS has mechanisms in place to prevent Service Principal Names (SPNs) that are duplicates in the forest. SPNs are automatically registered to computer objects in AD based on the services offered by the machine. There is a standard array of SPNs that every Windows client joined to the domain will register, for example. To see Microsoft's documentation with in-depth information on SPNs, go here:

http://technet.microsoft.com/en-us/library/cc961723.aspx

Issues can arise in Server 2012 R2 AD DS multi-domain environments, though, because by default Windows member machines (both clients *and* servers) will register a particular SPN that is **not** unique to the forest. That SPN is:

`HOST/machineName`

So if you have a machine named **TeacherComp**, then it'll register a SPN of:

`HOST/TeacherComp`

The problem here is that while other SPNs will frequently register with the FQDN (e.g. HOST/TeacherComp.providence.ketsds.net), this one does not. Thus while "TeacherComp" is a unique name at the domain level that does not guarantee it is a unique name at the forest level. While having non-unique names in different domains is fine – AD only prevents you from creating non-unique names within a domain – the fact that the SPN will not be unique across the forest is the problem. So while computers in different domains can have the same

name, the solution to the SPN problem is to make the names unique. In fact, this is part of the reason why Microsoft's recommendation is to make all computers in the forest uniquely named even across different domains:

https://support.microsoft.com/kb/909264?wa=wsignin1.0

From the above article:

> "*Use a unique name for every computer in your organization. Avoid the same computer name for computers in different DNS domains.*"

It's worth noting that this new uniqueness requirement will not prevent already established machines from continuing to function. For example, if a computer in **providence.ketsds.net** is named **LabComputer1** and a computer in **harrodsburg.ketsds.net** has the same name (which is entirely possible if both were joined to the domain prior to the Windows Server 2012 R2 AD DS upgrade), both will continue to operate normally. Problems will occur, though, if at any point in the future one of them is removed from the domain and then rejoined with the same name. In that instance, the name will need to be changed to something unique for the join to be successful. Likewise, joins for any brand new machines will also need to be unique.

For additional technical information, Microsoft publishes the following page regarding SPN uniqueness:

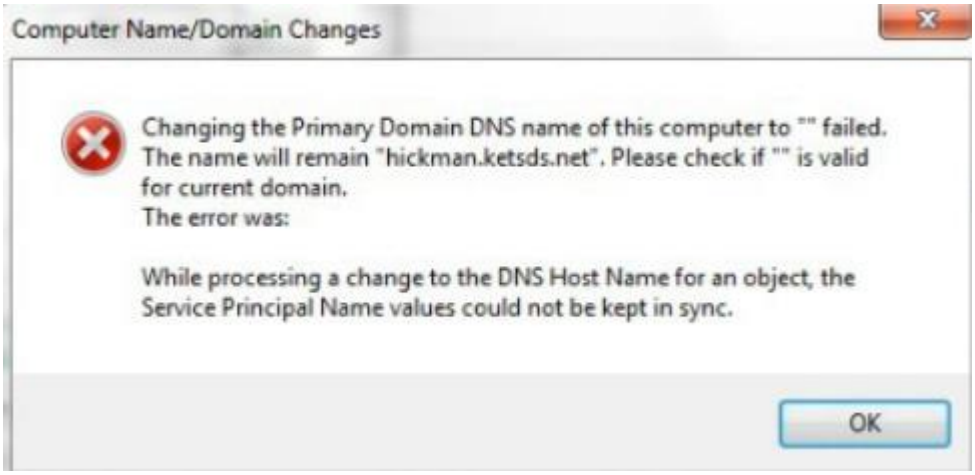
http://technet.microsoft.com/en-us/library/dn535779.aspx

3.1.6.2. Identifying the Problem

Unfortunately, the error message received when experiencing a problem with uniqueness is not particularly clear. When joining/rejoining a machine to the domain via the GUI, first the information is entered normally:

The standard message welcoming you to the domain is actually still displayed:

Immediately after clicking **OK**, though, a new error appears because the machine has now tried to register the SPN discussed in the section above and it fails:



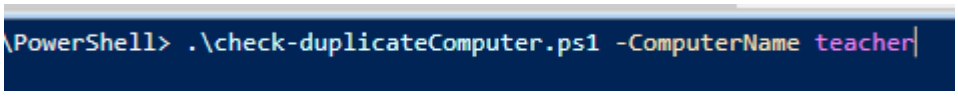
At this point, the machine will still not be successfully joined to the domain, and local credentials will need to be used to rename the machine and attempt the join procedure again. Likewise, a lingering computer object with the old name will remain in the **Computers** container of AD and need to be manually removed.

3.1.6.3. Preemptively Checking Uniqueness

In order to help alleviate some of the headaches with checking uniqueness, the MADS team has created a script to help. If district admins see the error message above when joining a machine or simply wish to see if a name is available prior to doing a join, the script can be utilized to see if that name already exists in the forest. The script is for PowerShell and can be accessed here:

http://1drv.ms/1tME5fN

The only input required for the script is the name to be checked. This can be passed either via the **–ComputerName** parameter or by specifying no parameters and letting the script prompt for the name:



```
PS C:\Users\jfabry\OneDrive\Documents\PowerShell> .\check-duplicateComputer.ps1
Importing AD module... Please wait.

Enter the computer name to query: teacher
```

Once the name has been entered, if there are no existing instances of it the script will return that in a message to the screen:

```
PS C:\Users\jfabry\OneDrive\Documents\PowerShell> .\check-duplicateComputer.ps1
Importing AD module... Please wait.

Enter the computer name to query: notaname
Finding your site...

There were NO results found. You can proceed with using: notaname.
```

If any instances are found, though, the script will return a warning message along with information on the domain or domains currently holding an object with that name:

```
PS C:\Users\jfabry\OneDrive\Documents\PowerShell> .\check-duplicateComputer.ps1 -ComputerName teacher
Importing AD module... Please wait.

Finding your site...

There were results found! You can't use teacher!

This machine name is being used in the following domain(s):
westpoint
butler
```

Prerequisites for the script include running it from a domain-joined machine while logged in with a domain account; local accounts will not work. The script queries the closest Global Catalog, so the user running it must be authenticated. Finally, the script leverages the AD PowerShell module, and so that must be installed on the machine executing it. The AD PowerShell module is attainable by installing the RSAT as described in Section 3.1.

3.1.6.4. Naming Recommendations

As a way to ensure uniqueness, KDE recommends including the 3-digit district number in all machine names in each domain. For example, instead of using **ElemLab1**, districts could use:

496ElemLab1

Or:

e496ElemLab1

Ultimately, however, this is simply a recommendation. Districts can use whatever scheme they are the most comfortable with as long as the name is unique across the forest.

3.1.7. AD Recycle Bin

3.1.7.1. Recovering Deleted Objects

With the completion of moving AD DS from the previous mode to Server 2012 R2 Native mode, the AD Recycle Bin has been enabled in each domain. The Recycle Bin offers the ability to quickly and easily have deleted users, computers, groups, OUs, etc. restored *without* the need for a time-consuming authoritative restore of the directory.

However, restoring deleted AD DS objects out of the AD Recycle Bin requires administrative permissions above what can be delegated to district IT staff. As a result, any need to have objects restored must be submitted as a ticket via the KETS Service Desk so that MADS can perform the recovery. It is worth mentioning that, unlike the authoritative restores which were previously required, recovering objects from the AD Recycle Bin is a quick process that can be done for both individual objects and bulk objects.

3.1.7.2. Technical Background

Deleting items from AD DS does not result in the immediate removal of the objects. Prior to the upgrade to Server 2012 R2, deleting an object caused it to be sent to the hidden **Deleted Objects** container that exists in each domain. All of the attributes would be stripped from the object except for a small handful, and the object would remain in this state for 180

days. This is done so that all of the other domain controllers will absolutely be guaranteed to have enough time to replicate the change. After 180 days, the object is actually removed from the directory.

With the Recycle Bin enabled, deleting an object still sends it to the hidden **Deleted Objects** container, but the objects there exist at two different levels. The first level marks the object as being *deleted*. The object still has all of its attributes and can be restored easily. The object will remain in this state for 180 days. After 180 days, if the object hasn't been restored then it moves from a *deleted* state to a *recycled* state. At that point, the object will have its attributes stripped, just like it did prior to enabling the Recycle Bin. The object will exist in this state for an additional 180 days prior to then being removed from the directory.

Microsoft's official documentation on the Recycle Bin is located here:

http://technet.microsoft.com/en-us/library/dd392261(v=WS.10).aspx

Microsoft has also published more detailed documentation on the flow of objects through the Recycle Bin prior to being removed from the directory here:

http://blogs.technet.com/b/askds/archive/2009/08/27/the-ad-recycle-bin-understanding-implementing-best-practices-and-troubleshooting.aspx

3.1.8. Applications Pointing To AD

For applications that statically point to AD for anything districts should validate which servers (name and/or IP) they point to. Examples of these types of applications are Infinite Campus LDAP to AD, iboss Network Security, Moodle authentication to AD, Apple Open Directory, etc.)

A best practice is to point applications to a district's namespace (e.g. providence.ketsds.net) rather than a specific server. However, that is not always possible since some applications require an IP address or server FQDN; this is especially the case when dealing with applications that use LDAPS as those must point to the FQDN of a particular server. In those instances, all applications that are physically located in the district should

point to either the on premise DC1 or GC1. Applications that are hosted elsewhere (e.g. Infinite Campus hosted in Frankfort) should point to the new DC3 server. The naming scheme and IP address of the servers are as follows:

Common Name	Name	IP
GC1	ED*xxx*ADGC1	10.253.*y*.10
DC1	ED*xxx*ADDC1	10.253.*y*.12
DC3	ED*xxx*ADDC3	10.226.18.*y*

In the table above, ***XXX*** is the district number (e.g. Adair's is 001) and ***Y*** is the district IP octet (e.g. Adair's is 21.) The relevant information for configuring this specifically in Infinite Campus is available at the following URL:

http://education.ky.gov/districts/tech/sis/Documents/AD_Infinite_Campus_Integration_in_KETS_Env_Technical_Guide.pdf

3.2. DNS Administration

Each school district in Kentucky has several DNS zones. Some of the zones, such as ***districtName*.ketsds.net**, are Active Directory integrated zones. This means that the DNS records and other zone information are replicated amongst the domain controllers within that domain. There are also secondary zones, such as ***districtName*.kyschools.us**, which are only copied to the domain controllers; BIND servers authoritatively host those zones, and the domain controllers do zone transfers with those BIND servers to store a local copy of the zone for faster resolution within the district. It is impossible to make changes to the zone from the domain controllers, though; they provide a read-only copy of the zone. For the purposes of this document, only the AD integrated zones will be discussed.

At a domain-specific level, each domain has two AD integrated zones:

- ***districtDomain*.ketsds.net**
- ***10*.in-addr.arpa**

The former is a forward lookup zone populated with A and CNAME records which are used to resolve domain joined workstations and servers from their FQDN (fully qualified domain name) back to an IP address. It also contains SRV and TXT records, to assist back-end processes with locating services (eg. KMS server, Kerberos server, etc.)The latter is a reverse lookup zone populated with PTR records that are used to resolve IPv4 addresses back to FQDNs.

District staff do not have administrative rights to DNS, so using the dnsmgmt.msc snap-in is not possible. Any modifications or troubleshooting with DNS will require a ticket with the KETS Service Desk. However, this section of the document will discuss several relevant pieces regarding how DNS is configured to make understanding the environment and submitting tickets easier.

3.2.1. Tertiary DNS Entries

By default, all DHCP scopes configured for the districts give the local DC1 and GC1 as the DNS servers for a client. However, some DHCP clients have the ability to accept a tertiary DNS server as well for use in instances where the first two are unavailable. For those clients it is possible to point them to the DC3 server in Microsoft Azure. This would allow those clients to continue communicating with Active Directory and resolving DNS queries even in the instance that the local domain controllers are unavailable.

Districts wishing to take advantage of this should open a ticket with the KETS Service Desk. If the district does not want or cannot have the tertiary DNS entry present on all scopes as a server option then which DHCP scope(s) need to have the modification applied should be specified.

3.2.2. DNS Scavenging

DNS scavenging is the process by which old or stale DNS records are removed. This is important because most DNS records in the KETS AD integrated DNS environment are not static records; they are dynamic records. This means that when a client machine leases an IP address via DHCP – which is discussed in the next section – that client will automatically register a DNS record with that IP address. Periodically, the client may either 1.) switch to a different IP address – especially if it moves to a different location in the district operating on a different subnet – and need to update the record or 2.) simply let DNS know that it is still using the same IP address, meaning the timestamp on the record needs to be modified. If neither of these happens, DNS will automatically delete the record after a certain period of time.

There are a lot of pieces at play, however, which determine both how quickly the record is deleted and how readily a client can have an existing DNS record updated to reflect a new IP address. There are two key parts to this: DNS server scavenging and DNS zone aging. Microsoft provides a very detailed write-up of the process here, and this document will provide additional details as they relate to KETS:

http://blogs.technet.com/b/dougga/archive/2012/02/09/it-takes-two-dns-scavenging.aspx

3.2.2.1. DNS Server Scavenging

For DNS records to be scavenged, a DNS server in the zone must be set to carry out the task at a set interval. This setting does **not** replicate to the other DNS servers, and generally speaking it is a best practice to only have one server per AD integrated DNS zone carrying out a scavenging task; if multiple servers are trying to run scavenging they will simply compete with one another which is unnecessary since the actual deletion of stale records will be replicated to all of them.

In the KETS environment, the DC1 in each district is responsible for scavenging. It carries this out every 3 days. When it runs every 3 days, it will delete any DNS record with a timestamp that is older than the zone's refresh interval + no refresh interval, which will be

Restarting the DNS Server service will also reset the scavenging interval. This means that if a district's DC1 is restarted for any reason (WSUS patches, hardware replacement, etc.) the scavenging interval will be a full 3 days from the time the service comes back online again. For example, if 2.5 days have gone by since the last run of scavenging and the DC1 is rebooted, it will take an additional 3 days for

discussed in the next section. Records with timestamps older than that will be deleted while records with timestamps newer than that will be left alone; static DNS entries have no timestamp and thus are never touched by scavenging.

3.2.2.2. DNS Zone Aging

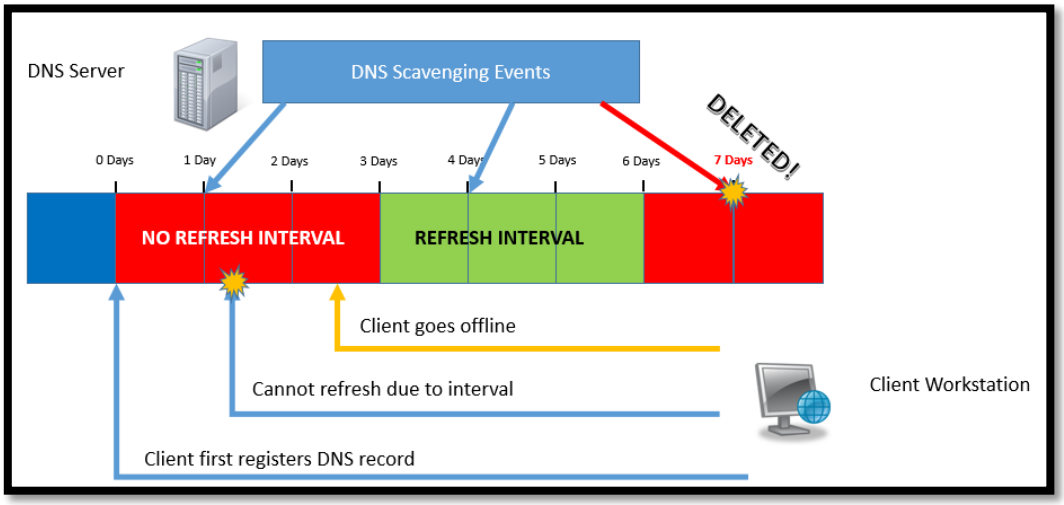
Aging is the process by which machines can and cannot update their DNS record. When a new client comes online, for instance, it will immediately register a record in DNS. That record will have a timestamp for when it was set, and the timestamp replicates to the other DNS servers in the domain. Once that has occurred, that machine will enter what is referred to as the **no refresh interval**. During this time period, the machine cannot update its DNS record even if it moves to another location in the district and receives a new IP. The reason for this is that any changes to the record's timestamp have to be replicated to all of the other domain controllers running the DNS service in that domain, which in the KETS environment is all 3 of them. Having a no refresh interval significantly cuts down on that replication traffic and improves domain controller performance. While the AD DS default for this interval is 7 days, in the KETS environment it has been moved to 3 days.

Once the 3 day no refresh interval has completed, the refresh interval begins. This is the window during which machines can either bump the timestamp to signify that it is still using the IP in the existing record or it can also completely update the record to a new IP address if that has changed during the no refresh interval. In the KETS environment, the refresh interval is also 3 days.

Once the refresh interval has passed, another no refresh interval begins and the cycle continues. If a machine does not update its record at all during the refresh interval, which usually happens because the machine is offline or is somewhere outside of the district, that record has now become eligible to be scavenged. That will happen when the DC1 performs its next run of periodic scavenging since the record's last timestamp is older than 6 days (refresh interval of 3 days + no refresh interval of 3 days.)

If a DNS record has the wrong IP because the address changed during the no refresh interval, opening a ticket requesting a manual run of DNS scavenging will NOT solve the issue because the record will not be old enough to be scavenged!

The process is demonstrated visually in the below image:



3.2.3. Static DNS Record Requests

Sometimes it may be necessary to create static DNS records rather than just relying on dynamic DNS registration. If a static record is necessary then it can be created by opening a ticket with the KETS Service Desk. The ticket should include the following information:

- **Requested hostname:**
  - **E.g. *myServer.district*.ketsds.net**
- **IP address**
- **Functional use of the device**
- **Whether or not a PTR record is needed for reverse DNS**

3.2.4. _VLMCS SRV Records

Along with the more common A, CNAME, and PTR records, there are also various SRV, or service, records in DNS. SRV records are typically automatically created by the application that needs them. Conflicts can arise, however, when multiple computers want to register the same record. A common occurrence for districts installing Windows Deployment Services (WDS) and/or Key Management Services (KMS) is for conflicts to arise with the _VLMCS DNS record.

While some services such as WDS like to use the _VLMCS record, it is not critical for them. For other services, such as KMS, registering the _VLMCS record is critical to the proper functioning of the system. Problems can also arise when, for example, WDS is installed on multiple servers for testing purposes, and the wrong server ends up registering the record.

While districts do not have the administrative rights necessary to view the _VLMCS record through the DNS Management snap-in, anyone can find the _VLMCS record in a given domain through the following query:

nslookup -type=srv _vlmcs._tcp.*district*.ketsds.net. ed*xxx*adyc1.*district*.ketsds.net.

So for example:

nslookup -type=srv _vlmcs._tcp.scott.ketsds.net. ed525addc1.scott.ketsds.net.

If the wrong machine has registered _VLMCS record, it has done so because it is running some service which makes the system think it needs to register the record. Prior to having the _VLMCS record deleted, district administrators should make sure that service is no longer running on the problematic machine. Failure to do so will likely result in the same incorrect machine re-registering the _VLMCS record once it has been deleted.

After querying to configure whether or not the proper machine holds the _VLMCS record, a ticket can be created with the KETS Service Desk to delete the record if necessary so that a different machine can register it.

3.3. DHCP Administration

DHCP is a lynchpin technology in many modern networks. Rather than forcing administrators to manually set and keep track of static IP addresses for client machines, DHCP allows those machines to automatically obtain IP addresses when they are connected to a network. While district administrators have no delegated rights in DHCP, they do have the ability to view current DHCP settings through the DHCP Management snap-in.

As described in section 2.1.3, most districts are given one Class B network, though in some instances larger districts are given multiple Class B networks. That Class B network is then further sub-divided into 16 Class C networks with a /20 subnet mask. Only 14 of the 16 are pre-created, leaving two potential /20 networks available to either be activated when the district expands or to be sub-divided into smaller /21, /22, /23, or /24 networks for small schools, bus garages, and other locations that need network connectivity without the full

allotment of addresses that a /20 network would provide. Furthermore, the pre-created /20 networks are disabled by default; a network can be activated for use through a request to the KETS Service Desk.

In the default /20 setup, the first 508 IP addresses are reserved for static assignment and are not made available to be leased dynamically. For example in a 10.x.16.0 /20 network, 10.x.16.1 – 10.x.17.254 are reserved for static use. The breakdown of how specific ranges within that span should be used is specified in the IP Numbering Standards document available by request from the KETS Service Desk. Outside of these and some IP addresses used for broadcasts that are excluded from DHCP, the default setup allows for **3556** leasable addresses per /20 scope.

3.3.1. DHCP Options

Along with an IP address, DHCP is also capable of delivering other pieces of networking information to clients. These are referred to as DHCP scope options, and they can be set at either the scope or server level. Several default options are always set at the server level:

Option Name	Vendor	Value	Class
006 DNS Servers	Standard	10.253.24.10, 10.253.24.12	None
015 DNS Domain Name	Standard	ANDERSON.ketsds.net	None
044 WINS/NBNS Servers	Standard	10.253.24.10, 10.253.24.12	None
046 WINS/NBT Node Type	Standard	0x8	None

- **DNS Servers**
  - **Always set to the district's local GC and DC**
- **DNS Domain Name**
  - **Will always be the district's domain of *district*.ketsds.net**
- **WINS Servers**
  - **Will be the district's local GC and DC**
- **WINS Node Type**

- **Sets WINS as a Hybrid node to query the WINS server and then do a broadcast if that fails**

On new DHCP scopes, the only default scope option will be for the scope's gateway. The gateway will always be the very first IP address in the scope (e.g. 10.x.16.1.)

Option Name	Vendor	Value	Class
003 Router	Standard	10.24.16.1	None

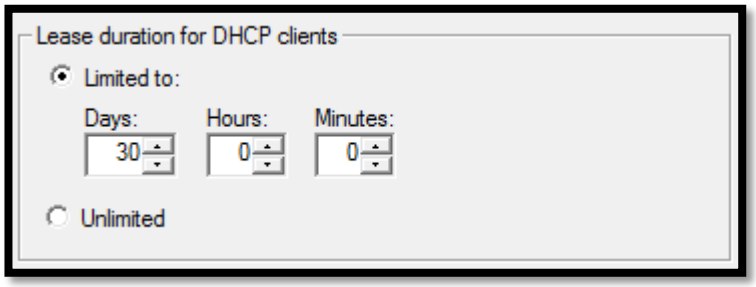
Additional information may be requested if the options are custom options from a vendor, as the options will have to be manually created by MADS and added to the DHCP server before their values can be set. A relatively common example of this is a request for Avaya options 176 and/or 242.

District administrators can request new scope and server options be added or modifications made to the existing ones. These can be requested via a ticket to the KETS Service Desk. Be sure to include:

- **The option to add/remove/modify**
- **The value that option should have**
- **If the option should be added as a scope or server option**
  - **If adding a scope option, specify which scope(s) it should apply to**

3.3.2. DHCP Lease Duration

Each DHCP scope has settings for how long clients should be allowed to lease an IP address. The KETS default is 30 days, though this value can be modified at the request of district administrators to any value. Many districts, for example, keep a 30 day lease duration on their wired networks while using a 1 day lease duration on wireless networks:



It is important to consider lease duration when activating new scopes. As was previously mentioned, the default in a /20 scope is to have 3556 addresses. The lease

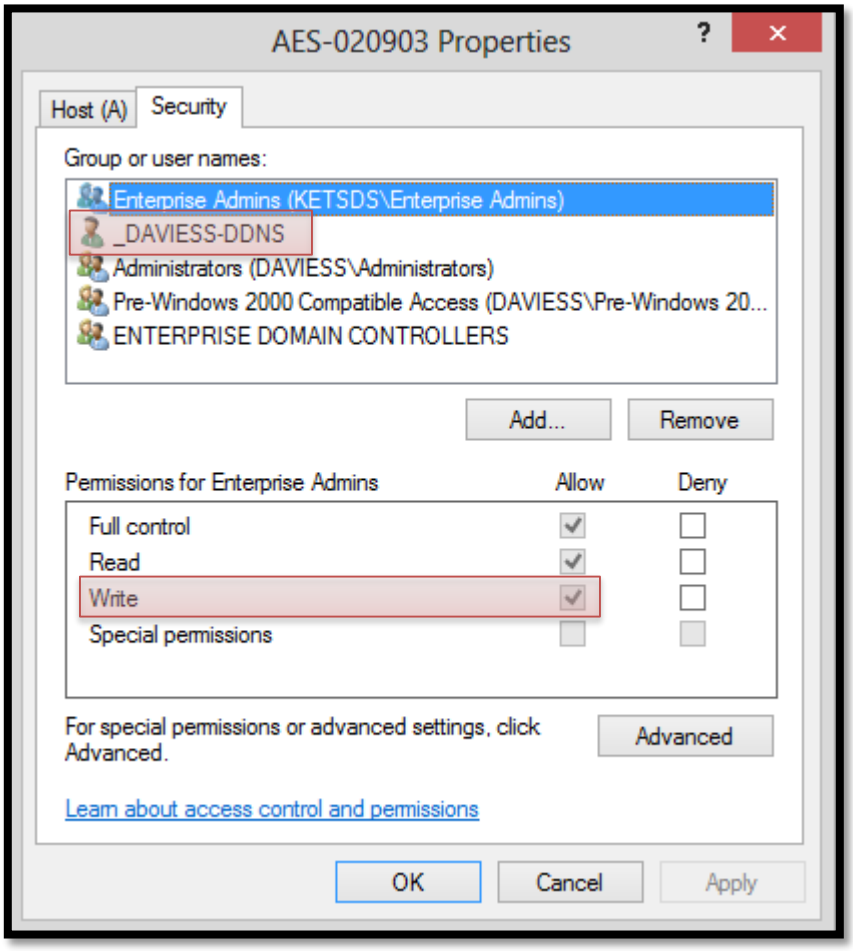
duration should be short enough that client machines physically taken out of the district will not hold a lease for too long of a time thus causing the scope to run out of available IPs to lease. For details on how the lease renewal process works, see the following Microsoft documentation:

http://technet.microsoft.com/en-us/library/cc958919.aspx

Should a scope reach capacity and no longer have available IPs to lease, new clients will not be able to get an IP address on that scope. At that time there are very limited options for solving the issue aside from letting adjusting the lease duration and then letting the current leases expire naturally.

3.3.3. DHCP/Dynamic DNS

Currently both local district domain controllers have Dynamic DNS configured in the DHCP server setup. This means that DHCP can create and maintain the DNS A record for a client that requests a lease. The current expected behavior for DHCP clients in the KETS environment is that once a client requests a lease from one of the two DHCP servers in a district that DHCP instance can create a corresponding DNS A record for the client using a Dynamic DNS service account that exists in each domain. The client DNS A record will then have an ACE (Access Control Entry) for the Dynamic DNS service account in its ACL (Access Control List) that allows the DHCP service to act on behalf of the client and update the DNS A record when the lease information for that client changes. This is useful not only for keeping DNS in sync with DHCP but also for automatically creating records for clients that cannot register DNS records on their own such as printers, or non-windows clients.



1 - DNS A RECORD ACL

However, there is a small limitation with using DDNS. If a client first obtains a DHCP lease, and then subsequently moves to a static IP address, the client does not have the rights on the DNS record to update itself. We primarily saw this issue occur with Servers, but can occur with Workstations as well.

3.4. Group Policy Administration

Group Policy provides an infrastructure through which administrators can have very granular control of both domain-joined computer systems and domain user accounts. It gives

administrators the ability to control everything from the desktop wallpaper a user sees when logging in to a domain-joined computer to the firewall and WSUS settings of a machine. Group Policy is a vehicle by which users can be provided the best/safest/most compatible working environment while simultaneously protecting them from accidentally changing it.

The GPMC can be installed via the same RSAT package used to get ADUC, ADAC, and other tools for use against AD DS.

Group Policy is assigned at the OU level. This means that when you create a GPO, you should keep in mind that you will be applying it to an OU rather than an AD Security Group, for example. It is also important to keep in mind that Group Policy is **not** applied to objects which are actually *containers*; only the OUs visible in the Group Policy Management Console (GPMC) receive Group Policy.

This can cause issues if, for instance, administrators leave newly domain joined computers in the Computers container since, as a container rather than an OU, it does not receive Group Policy. If there is a desire to have Group Policy applied to those machines, then it will be necessary to move them to an OU with the appropriate GPOs applied.

Microsoft provides authoritative documentation on Group Policy. For additional information, please visit the following:

http://technet.microsoft.com/en-us/windowsserver/bb310732.aspx

3.4.1. Group Policy Objects

Group Policy settings are contained within Group Policy Objects, or GPOs. These GPOs are stored in the Sysvol of each domain and replicate amongst the domain controllers in that particular domain. The GPOs are stored at the following location:

```
\\district.ketsds.net\SYSVOL\district.ketsds.net\Policies
```

The Scripts folder is not designed for storing massive files (e.g. the Office 2013 installer.) If there is a need to store such large files, it is recommended that district utilize their own file servers. This is especially important as each domain's tertiary domain controller is now stored in Microsoft's Azure cloud, and the Sysvol replication of unduly large files will result in a hefty utilization of storage and network resources.

At the same level as the **Policies** folder is the **Scripts** folder. This folder is designed for the storage of scripts and other *small* files that may need to be referenced by your GPOs.

3.4.2. ADM vs. ADMX Templates

The settings available for modification in a GPO are determined by the template files available. Template files come in two different flavors: **ADM** and **ADMX.** ADM files, with an extension of .adm, are a type of local template file. If you create a GPO using an ADM file, the GPO will store a copy of that ADM file in the Sysvol. This then allows any other administrators with the proper delegation, which will be discussed later, to modify the GPO even if they don't have the corresponding ADM file stored locally. If another policy is created that utilizes the same ADM file then another copy of the ADM file is stored in the Sysvol for that GPO.

ADMX files are a newer variant of the template with a .admx extension. Rather than being used locally, they are contained within what is called the **Central Store**. The Central Store is simply a folder created under the Policies subfolder in the Sysvol. With all of the ADMX files there, any GPOs that are created will reference the file in the Central Store rather than having a self-contained copy of the template. This can make for a more efficient use of space on the domain controllers since there are no template files being duplicated.

The Central Store cannot be used concurrently with ADM files. Once the Central Store has been created, any local ADM files will be ignored when using GPMC. Any administrators wishing to utilize the Central Store must understand that any specialized, local ADM files they may have will not be available for subsequent policies!

For a more detailed look on ADM and ADMX files, see the following:

http://technet.microsoft.com/en-us/magazine/2008.01.layout.aspx

For more information on the Central Store, review the following:

http://support.microsoft.com/kb/929841

3.4.3. Group Policy Delegation

When an administrator creates a GPO, the Domain Administrators group becomes the Creator Owner of the Group Policy object. By default, Domain Administrators can edit all GPOs in the domain.

The ability to create GPOs in a domain is a permission that is managed on a per-domain basis. By default only Domain Administrators, Enterprise Administrators, Group Policy Creator Owners, and SYSTEM can create new Group Policy objects. If the domain administrator wants a non-administrator or non-administrative group to be able to create GPOs, that user or group can be added to the Group Policy Creator Owners security group. Alternatively, you can use the **Delegation** tab on the Group Policy Objects container in the GPMC to delegate creation of GPOs. When a non-administrator who is a member of the

Group Policy Creator Owners group creates a GPO, that user becomes the creator owner of the GPO. That user can then edit and modify permissions on the GPO. However, members of the Group Policy Creator Owners group cannot link GPOs to containers unless they have been separately delegated the right to do so on a particular site, domain, or OU. Being a member of the Group Policy Creator Owners group gives the non-administrator full control of only those GPOs that the user creates. Group Policy Creator Owner members do not have permissions for GPOs that they do not create.

The right to link GPOs is delegated separately from the rights to create and edit GPOs. Be sure to delegate both rights to those groups you want to be able to create and link GPOs. By default, non-Domain Admins cannot manage links, and this prevents them from being able to use GPMC to create and link a GPO. However, non-Domain Admins can create an unlinked GPO if they are members of the **Group Policy Creator Owners** group. After a non-Domain Admin creates an unlinked GPO, the Domain Admin or someone else who has been delegated permissions to link GPOs to a container (e.g. DIST Support Admins members) can link the GPO as appropriate.

Creation of GPOs can be delegated to any group or user. There are two methods of granting a group or user this permission:

- **Add the group or user to the Group Policy Creator Owners group. This was the only method available prior to GPMC.**
- **Explicitly grant the group or user permission to create GPOs. This method is newly available with GPMC.**

You can manage this permission by using the **Delegation** tab on the Group Policy objects container for a given domain in GPMC. This tab shows the groups that have permission to create GPOs in the domain, including the Group Policy Creator Owners group. From this tab you can modify the membership of existing groups that have this permission, or add new groups.

Because the Group Policy Creator Owners group is a domain global group, it cannot contain members from outside the domain. Being able to grant users permissions to create

GPOs without using Group Policy Creator Owners facilitates delegating GPO creation to users outside the domain. Without GPMC this task cannot be delegated to members outside the domain.

If you require that users outside the domain have the ability to create GPOs, create a new domain local group in the domain (e.g. "GPCO – External"), grant that group GPO creation permissions in the domain, and then add domain global groups from external domains to that group. For users and groups in the domain you should continue to use the Group Policy Creator Owners group to grant GPO-creation permissions.

Adding a user to the membership of Group Policy Creator Owners and granting the user GPO-creation permissions directly using the new method available in GPMC are identical in terms of permissions.

It is recommended that districts make sure permission to modify existing GPOs is never limited to just one individual in the district. The MADS team has no bulk means by which to mass-modify GPOs to adjust the permissions, meaning that it will be left to the district staff to individually modify each GPO so that it can be changed by another administrator in the event that becomes necessary (e.g. due to vacations, staff turnover, etc.)

3.4.4. Common Questions

Setting IE10 and IE11 Proxy Settings Via Group Policy

The manner in which Proxy Settings are done via Group Policy changed with IE10 and IE11. The process to do this is detailed here:

https://staffkyschools-my.sharepoint.com/personal/john_fabry_education_ky_gov/Blog/Lists/Posts/Post.aspx?ID=2

Adding Sites To the Trusted Sites Zone Via Group Policy

The method for setting proxy settings via Group Policy for IE10 and IE11 shows an option for the Trusted Sites Zone, but it is grayed out. To manage this setting via Group Policy, utilize the following:

https://staffkyschools-my.sharepoint.com/personal/john_fabry_education_ky_gov/Blog/_layouts/15/start.aspx#/Lists/Posts/Post.aspx?ID=3