IBM Security Secret Server

Discovery Guide

Contents

Introduction

In Discovery, you create Discovery Sources that are used to scan for machines, local accounts and dependencies on Active Directory, Unix systems, and VMware ESX servers. Discovery is easy to set up and provides a great range of customizations for specific network requirements.

HOW DISCOVERY WORKS

Active Directory / Local Windows Accounts / Service Accounts

Secret Server queries Active Directory domains to obtain the list of Organizational Units (OU's) and Windows computers on the Domain. These OU's and computers are brought into the Secret Server database. Secret Server then attempts to connect to each computer and query for the following:

- **Local Accounts**   Local Windows accounts
- **Domain Accounts**   Active Directory User Accounts
- **Windows Services**   Windows services run by Active Directory accounts
- **Scheduled Tasks**   Windows scheduled tasks run by Active Directory accounts
- **IIS Application Pools**   IIS application pools run by Active Directory accounts

Unix accounts

Secret Server is given a list of IP address ranges on the network to scan for. It will look for computers that are listening on the specified ports (default is 22). Secret Server will then attempt to use DNS to resolve the list of IP's that are found with the goal of providing a more easily recognizable computer name. All computers found will be saved in the Secret Server database. Secret Server then attempts to connect to each computer using the provided credentials and query for the list of users on the target system.

VMware ESX accounts

Secret Server is given a list of IP addresses or computer names that correspond to ESX and/or ESXi servers. Secret Server then connects to each server using the provided credentials to query for the list of users on the target system.

WHY USE DISCOVERY?

Quick network credential import

By using Discovery, you use Secret Server to offset the burden of keeping track of computers and accounts on your network. This can be beneficial when getting started by bulk discovering and importing accounts, as well as having Secret Server find accounts and create Secrets whenever a new machine or account is provisioned moving forward.

Security against backdoor accounts

When Secret Server is configured to discover new accounts, it provides added protection by checking your network regularly for new accounts, which Secret Server can take over by importing and resetting their passwords to values that meet your security policy. Consequentially, if someone is setting up backdoor admin accounts on the network, they won't be able to use those accounts very long before they are imported into Secret Server and their passwords are changed.

DISCOVERY TYPES

Active Directory Discovery

Active Directory Discovery allows Secret Server to scan for Active Directory (AD) machines, Active Directory user accounts, local Windows accounts and dependencies on an AD domain. Secret Server will first discover machines from your domain; next, each machine is scanned for local Windows accounts and dependencies. By default, you can have Secret Server scan for local accounts, domain accounts, scheduled tasks, Windows services, and IIS application pools. You can find additional accounts and dependencies by creating PowerShell scanners. PowerShell scanners are an advanced topic described in the **Extensible Discovery** section.

Unix Discovery

The Wizard will help you get Unix Discovery configured in 3 simple steps: (1) name the Discovery Source, (2) define the host ranges of the IP addresses that you would like to scan, and (3) choose a Secret to use as credentials for scanning. The default command sets that Secret Server ships with will work to discover machines and accounts in most Unix environments.

By default, the **Find Non-Daemon Users (Basic Unix)** command set will be used. If the built-in account should be discovered, the Discovery Source must be updated to use the **Find All Users (Basic Unix)** command set. New command sets can be created by clicking **Configure Command Sets** when on the Discovery Sources list page.

ESX Discovery

The Wizard will help you get ESX/ESXi Discovery configured in 3 simple steps: (1) name the Discovery Source, (2) define the IP addresses of the ESX/ESXI servers, and (3) choose a Secret to use as credentials for scanning.

Generic Discovery Credentials

The Generic Discovery Credentials Secret type can store a simple username and password pair to be used for Unix or ESX Discovery. It is only intended for use in Discovery and is incapable of Remote Password Changing.

Configuring Standard Discovery

Note Discovery Requires Secret Server Professional Edition.

CREATE DISCOVERY SOURCE

1. Go to the **ADMIN** menu and select **Discovery**.
2. Click Edit Discovery Sources.
3. You will be taken to the Discovery Sources page where any existing Discovery Sources will be displayed. From this page you can create new Discovery Sources or edit your existing Sources.

Note If you have upgraded from an earlier version of Secret Server and have created an Active Directory Domain within Secret Server, a corresponding Discovery Source will be displayed on this page. If Discovery was not enabled on that domain, the Discovery Source will be "Inactive".

4. To add a new Discovery Source, click **Create New**. To edit an existing Discovery Source, click its name.
5. A windows will appear prompting for the type of Discovery Source to create: **Active Directory**, **Unix**, or **VMWare ESX/ESXi**.
6. Select the type of Discovery source you wish to create, then click **OK**.
7. You will be taken to the Wizard for creation of the chosen Discovery source type. Each Wizard will guide you through setup of the particular type in a series of steps. The first step of each Wizard is an overview page which describes Discovery Source. Clicking **Skip Wizard** allows you to skip the wizard completely and enter all settings at once on a single page. Clicking **Cancel** at any step of the wizard will end the wizard and clear any entered data. You can also navigate away from the creation wizard at any point using the navigation bar, and when you next return your progress will have been saved.

Figure 1 ADMIN menu

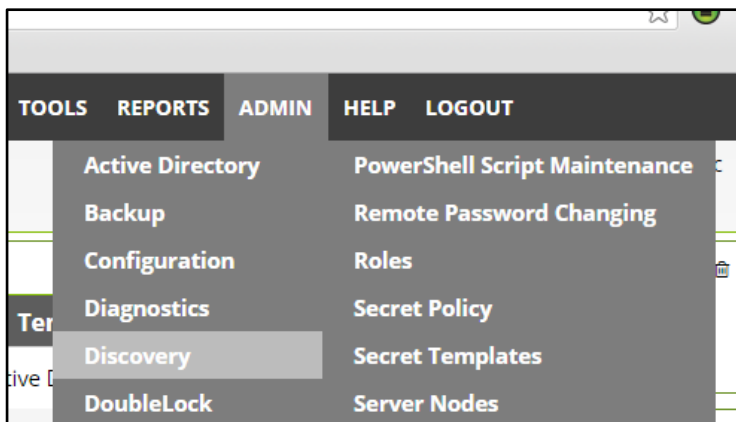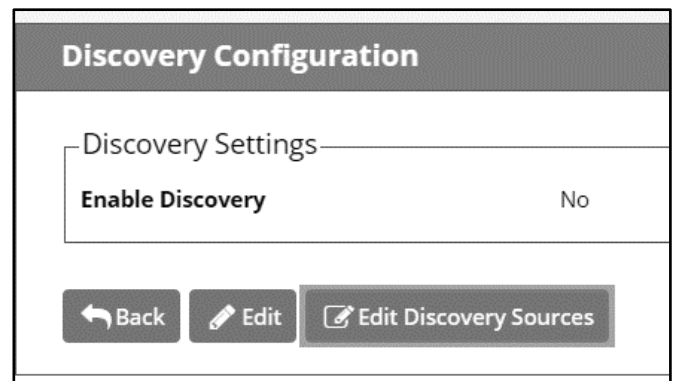**Figure 2** Edit Discovery Sources button

Figure 3 Discovery Sources. If the **Show Inactive** check box were not selected, the Active Directory source (fullyQualified.domainName.com) would be hidden.

Figure 4 Create New Discovery Source

Figure 5 AD Wizard Overview

ACTIVE DIRECTORY DISCOVERY SOURCE

The first step of creating an Active Directory Discovery Source will briefly summarize what an AD Discovery Source is. This is the only step that allows you to skip the creation wizard.

A Discovery Source created with this wizard will automatically have Discovery enabled for the entire domain. This setting can be changed by editing the Discovery Source after completing the wizard, or by clicking **Skip Wizard** from the start and setting it up manually. Otherwise, click **Next** to continue.
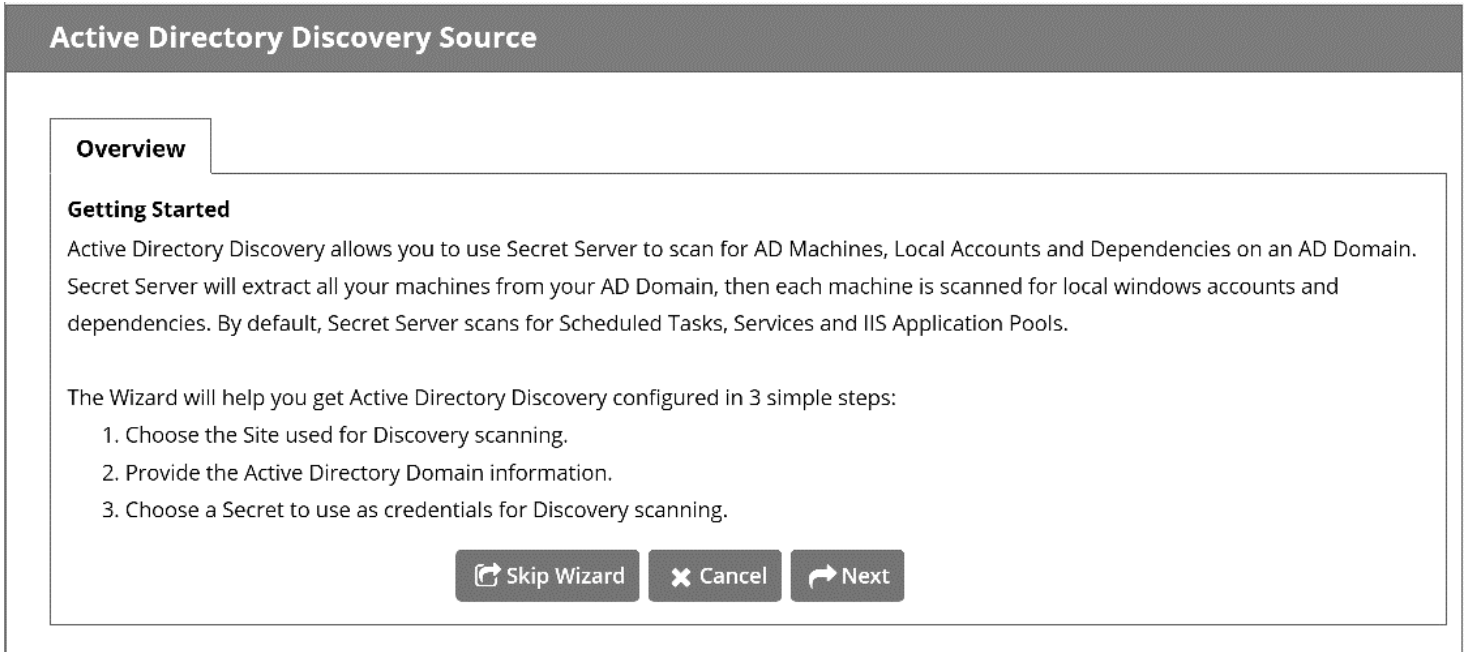
Figure 6 AD Discovery Source Overview

Site

Specify which Site with be used for this Discovery Source. If Distributed Engines are setup, a pull-down menu will show all active Sites. If no Distributed Engines are setup, the Site selection will default to Local, no changes will be made on this tab, and this tab will not be accessible.
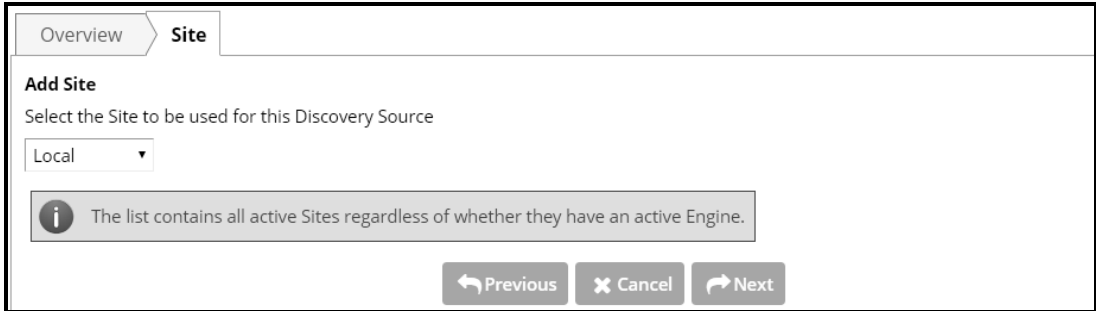
Figure 7 AD Wizard - Site

Settings

Specify domain name and credentials for Active Directory domain synchronization before clicking **Next** to continue.

- **Domain Name**   Fully-qualified domain name
- **AD Synchronization Secret**   A secret containing credentials that can be used to scan the domain for Active Directory users and groups. If you do not have a secret containing appropriate credentials, click **Create New Secret** to create a new secret then click **No Secret Selected** to select the secret you just created.

Note The information you enter here will attempt to authenticate. Secret Server must have access to the domain provided, and the account credentials must work.

Figure 8 AD Wizard - Settings

Discovery Credentials

The final step in Active Directory Discovery Source creation is optional. You may choose to select a secret here whose credentials will be used to authenticate when performing Discovery. If they are not specified, the secret from the previous wizard step will be used.

If you wish to select a secret, click **No Secret Selected**, and you will be given the opportunity to select a valid secret.

If the desired credentials do not yet exist in Secret Server, clicking **Create New Secret** will allow you to create a secret in a new tab which will then be immediately available in the picker.

Clicking **Finish** will conclude the wizard and return you to the Discovery Sources page where you will see the newly created Active Directory Discovery source.

Figure 9 AD Wizard – Discovery Credentials

Figure 10 AD Wizard – Secret Picker

UNIX DISCOVERY SOURCE

The first step of creating a Unix Directory Discovery Source will briefly summarize what a Unix Discovery Source is. This is the only step that allows you to skip the creation wizard.

A Discovery Source created with this wizard will have the Command Set, Ports, Max TCP Connections, and Parse Format settings applied with default values. These values can all be changed by editing the Discovery Source after finishing the wizard, or by clicking **Skip Wizard** from the start and entering the information manually (default values will be auto-filled for your convenience). Otherwise, click **Next** to continue.

Figure 11 Unix Discovery Source Overview

Discovery Source Name

Choose a name for your Unix Discovery Source, then click **Next**.

Figure 12 Unix Wizard – Discovery Source Name

Scan Range

Host Range of IP addresses to be scanned – The IP address range to scan for Unix computers. Multiple entries should each be on their own line. Host name entries are also allowed. The more precisely you are able to specify the actual ranges you wish to Discover Unix machines on, the faster the Discovery scan will run.

Figure 13 Unix Wizard – Scan Range

Site

Specify which Site with be used for this Discovery Source. If Distributed Engines are setup, a pull-down menu will show all active Sites. If no Distributed Engines are setup, the Site selection will default to Local, no changes will be made on this tab, and this tab will not be accessible.

Figure 14 Unix Wizard – Site

Discovery Credentials

Select one or more Secrets by clicking **Add Secret** and using the Secret Picker. These Secrets' credentials will be available for use when scanning for Unix machines and local accounts.

Note If multiple credentials are supplied, Discovery will go down through the list attempting each credential until it either has a successful authentication or had tried all provided accounts. This loop is donse for each computer.

If the desired credentials do not yet exist in Secret Server, clicking **Create New Secret** will allow you to create a Secret in a new tab which will then be immediately available in the picker.

Clicking **Finish** will conclude the wizard and return you to the Discovery Sources page where you will see the newly created Unix Discovery Source.
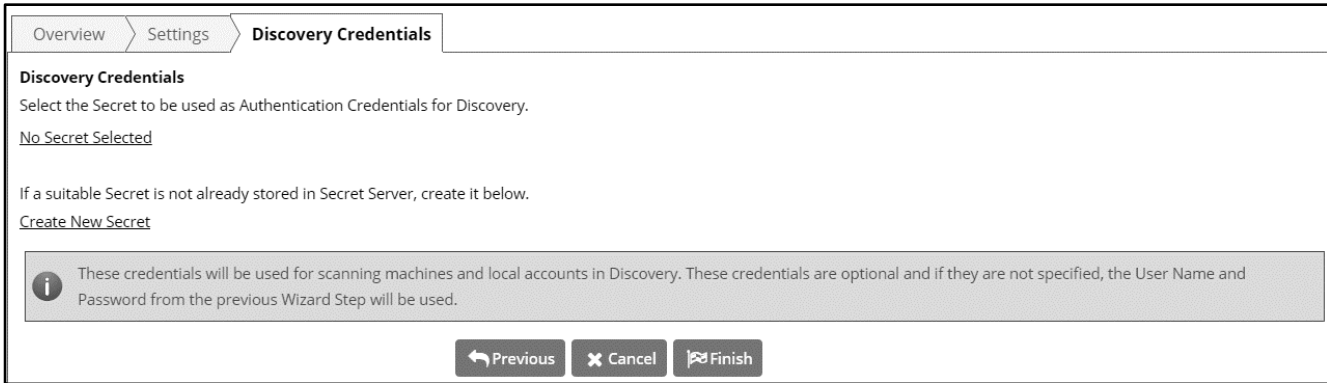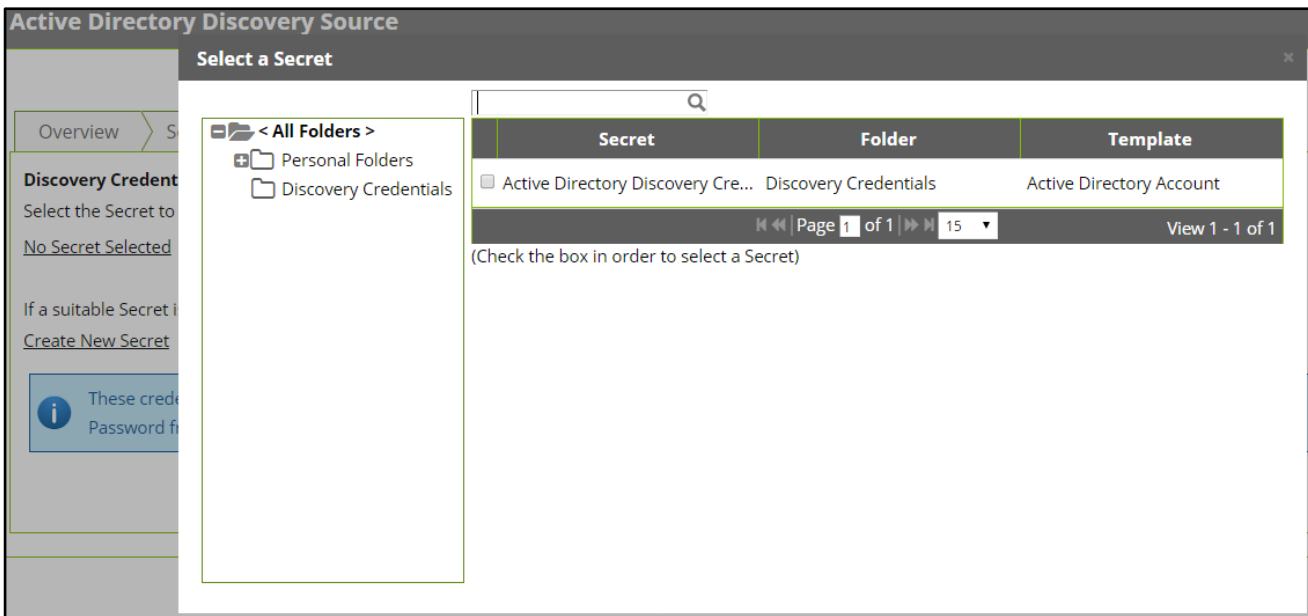

Figure 15 Unix Wizard – Discovery Credentials

ESX DISCOVERY SOURCE

The first step of creating a VMWare ESX/ESXi Discovery Source will briefly summarize what an ESX Discovery Source is. This is the only step that allows you to skip the creation wizard.

A Discovery Source created with this wizard will have the Port set to a default value. This value can be changed by editing the Discovery Source after finishing the Wizard, or by clicking **Skip Wizard** from the start and entering the information manually (the default value will be auto-filled for your convenience).
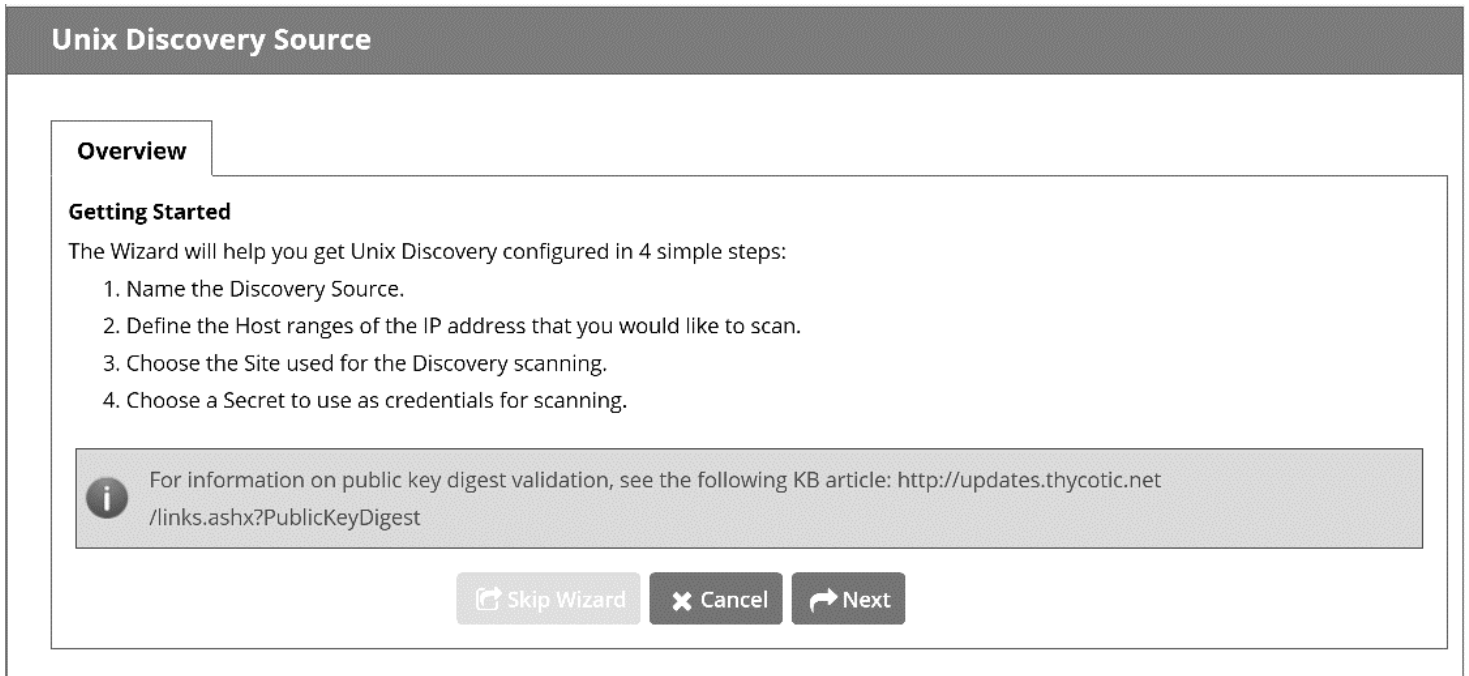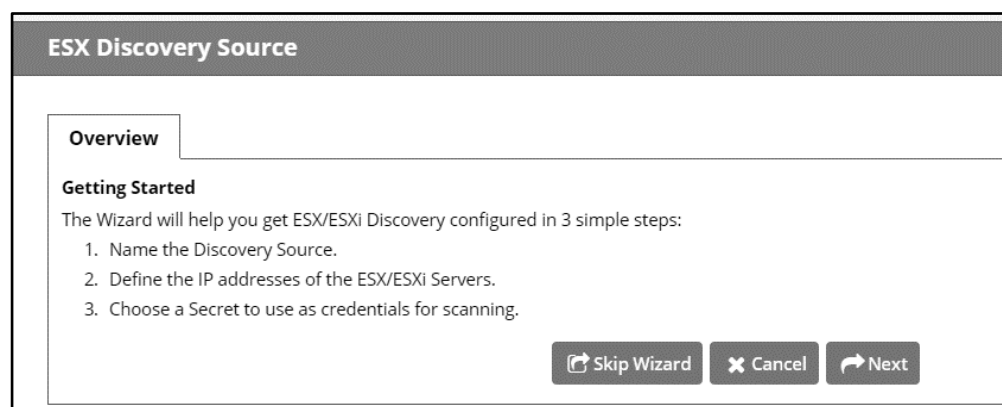

Figure 16 ESX Discovery Source Overview

Discovery Source Name

Choose a name for your ESX Discovery Source, then click **Next**.

Figure 17 ESX Wizard – Discovery Source Name

Servers

Enter the individual IP addresses or DNS names to scan for accounts. Each entry should be on its own line and must be a specific address rather than a range.
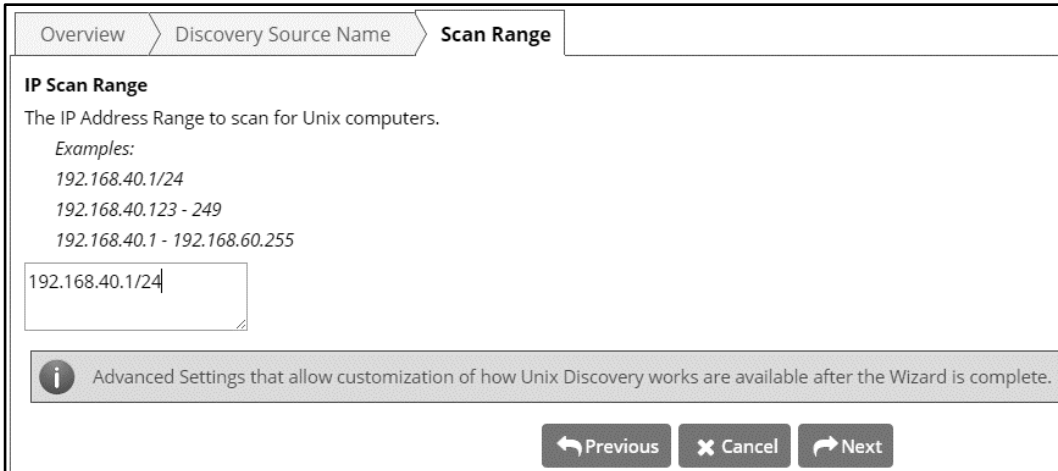

Figure 18 ESX Wizard - Servers

Site

Specify which Site with be used for this Discovery Source. If Distributed Engines are setup, a pull-down menu will show all active Sites. If no Distributed Engines are setup, the Site selection will default to Local, no changes will be made on this tab, and this tab will not be accessible.

Figure 19   ESX Wizard - Site

Discovery Credentials

Select one or more Secrets by clicking **Add Secret** and using the Secret Picker. These Secrets' credentials will be available for use when scanning for VMware ESX/ESXi accounts.

If the desired credentials do not yet exist in Secret Server, clicking **Create New Secret** will allow you to create a Secret in a new tab which will then be immediately available in the picker.

Clicking **Finish** will conclude the wizard and return you to the Discovery Sources page where you will see the newly created ESX Discovery Source.
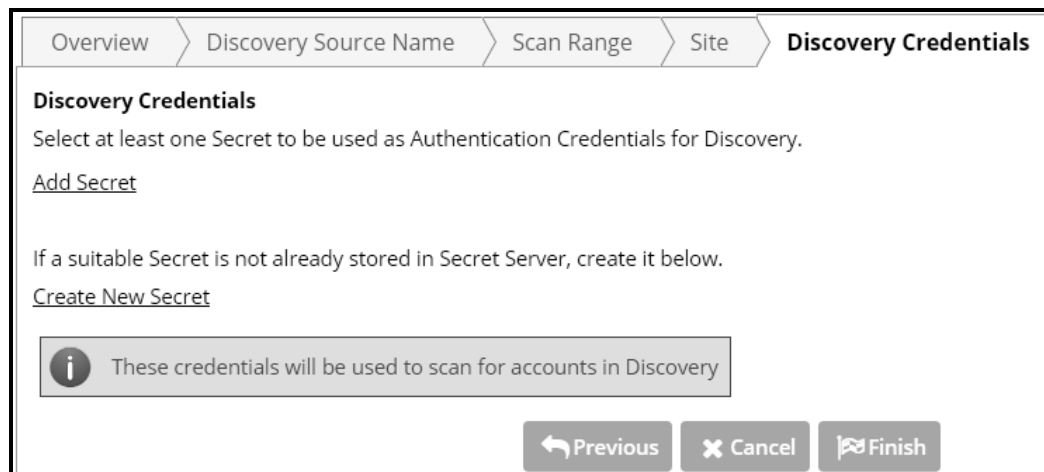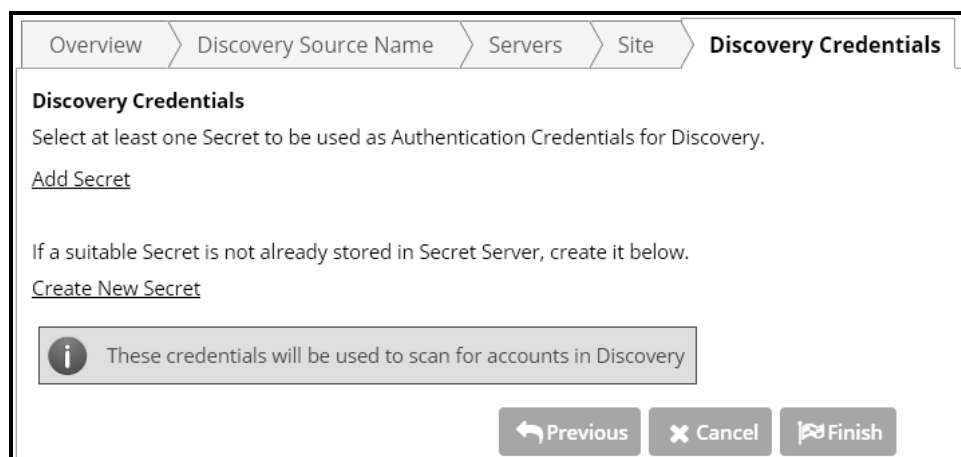

Figure 20   ESX Wizard – Discovery Credentials

MANUAL DISCOVERY SOURCE CREATION AND EDITS

Skipping the wizard at the start of Discovery Source creation, or editing an already existing source, allows you to configure every setting of the Discovery Source from a single page. This option is only available for Active Directory discovery sources. For Unix and VMWare ESX/ESXi you must go through the wizard.

Active Directory Source

Adding a new domain as a Discovery Source will also allow it to be used as a Synchronization Source.

> **Note**   If a domain was added as Active Directory Synchronization Source within Secret Server, but Discovery was not initially enabled, the domain will be listed as an "Inactive" Discovery Source. On the Discovery Sources page, check the box to "Show Inactive and Disabled" to see this domain.

Figure 21 Active Directory settings

ADVANCED DISCOVERY SETTINGS

Discovery Sources

Discovery Sources are definitions for scanning computers on your network for Active Directory domain-joined computers, Unix computers, and VMware ESX/ESXi machines.

Discovery sources share the following basic fields:

- **Discovery Source Name** Define the Name of your Discovery Source. Every Discovery Source must have a unique name. If you are using Active Directory then the domain name becomes the Discovery Source name.
- **Active** Activate this Discovery Source for scanning. Active Discovery Sources will be scanned at the Interval defined for Discovery. If you have multiple Discovery Sources, the Discovery Source with the most un-scanned computers will be scanned first.
- **Site** If you are using Distributed Engines, you can select the site that will run Discovery for the source.

Secret Server supports a number of different types of Discovery Sources which will each have their own specific settings.

Active Directory Discovery Source

Credential Secrets

- **Use Sync Credentials** The existing credentials use for domain syncing of users and groups will be used for Discovery.
- **Link a Secret** A selected Secret will be used as the credentials for Discovery Scanning. These credentials must have the proper rights to scan the remote machines. **Discovery Secret** will appear and allow you to select a Secret for use with Discovery.
- **Machine Resolution Type** The method by which Discovery will attempt to identify machines. The default is to use the machine and fully-qualified name. The other option is to use the machine name only.

Specific OUs

Active Directory Discovery can be scoped to run just in specific OU's. This can be enabled by selecting the **Specific OU** option for **Enable Discovery.** Save and then go to the Specific OU's tab.

OU's can be added from the autocomplete. Once added you can set a specific Site and Secret to run Discovery in that OU by clicking the icon to the right of the OU name.
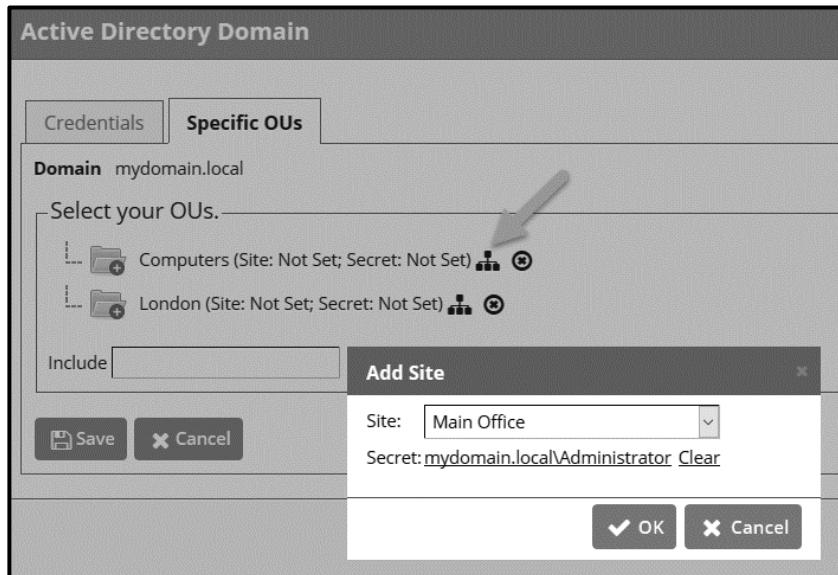
Figure 22 Setting a specific site and secret on an OU

- **Site** This is the Distributed Engine Site that will run Discovery for this OU. For example, an OU in a DMZ might need its own Site to limit firewall rules needed to scan machines.
- **Secret** A selected Secret will be used as the credentials for Discovery Scanning in this OU. Scoping Secrets by OU allows splitting out where Secrets used by Discovery need elevated rights.

Scanner Settings

This tab defines which scanners are used on this discovery source. Secret Server automatically adds built-in scanners for Organizational Units, Computers, and Local Accounts. Secret Server supports multiple scanners at each level except the first (Host Ranges), but in most cases the defaults provided should be sufficient.

By default, no dependency scanners are enabled on a discovery source. If you want to scan for Windows Services, Scheduled Tasks, or Application Pools, click **Add New Dependency Scanner** and click the "+" icon next to the dependency scanner you want to add. You can add as many dependency scanners as you need.

Figure 23   Default Active Directory Scanner Settings

Unix Discovery Source

This option will find all machines and local accounts on a set of manually defined host ranges for Unix machines that can be successfully accessed with SSH.

Host Range

- **Manual input** Define how you would like to find computers you would like to manage. This needs to be a valid IP address range. Multiple IP address ranges are allowed, but not overlapping IP address ranges on the same Discovery Source. There should be one IP address range per line in this field.

Find Machines

Define how you would like SSH Discovery Scanner to find machines in your host ranges. Each machine is scanned using SSH and the settings defined on the scanner. To obtain more information from the machine scan, use the default custom commands and authentication and Discovery can return the OS type of the Unix machine. To configure the settings for a scanner, click the pencil icon next to the scanner. The configuration options available for a UNIX machine scanner are:

- **Credentials Secret** Select the credentials to be used to scan with. These credentials may be Generic Discovery Credentials or a Unix Account (SSH) type Secret. You can add multiple accounts here.
- **Command Set** The command set is defined on the discovery scanner. To change command sets, create a new item scanner. Command sets are Unix commands that will be used to gather information from them machine when it is scanned for Discovery. Commands are only run on machines when authentication is enabled and a credential Secret is added to the Find Machine Settings. The default command set is **Find Machine (Basic Unix)**. This command set returns the OS.
- **Ports** This is a comma-separated list of port values (1-65535). SSH generally uses port 22.
- **Max TCP Connections** Enter the thread limit for scanning. This determines how many concurrent threads will run when scanning your network.
- **Attempt Authentication** Required to run commands on the machines being scanned. The credentials supplied by the Secret will be used to access each machine during the scan, if the credentials are correct, the custom commands will be run to extract the OS information from the machine.

For more information about discovery scanners, see the **Scriptable Discovery** section.

Find Local Accounts

Use SSH to Find Unix Local Accounts on machines discovered in your host ranges.

- **Credentials Secrets** Select the credentials to be used to scan with. These credentials may be Generic Discovery Credentials or a Unix Account (SSH) type Secret. You can add multiple accounts here.
- **Command Set** The command set is defined on the discovery scanner. To change command sets, create a new item scanner. Command sets are Unix commands that will be used to gather information from them machine when it is scanned for Discovery. Commands for Local Account Discovery are always enabled and a Credential Secret is required for the Find Local Account settings. The default command set is **Find Non-Daemon Users (Basic Unix)**. This will extract the non-built-in Unix users.
- **Ports** This is a comma-separated list of port values (1-65535). SSH generally uses port 22. The default port used when attempting to scan a machine for users. This will be overridden by a specific port found during machine scanning.

- **User Regex Format** This regular expression determines which lines of text received during the scan are actually valid for user parsing. The match groups in the regular expression should correspond to the comma-separated items in the Parse Format.
- **Parse Format** The Parse Format determines the order of the values that will be retrieved during a scan. If the parse names match the fields defined on the Secret imported, they will be populated from the data collected on the scan.
- **Newline Separator Character** This character will divide the lines in the output encountered during a scan.

Note If you use an IP address (instead of a host name) as the basis for a Secret and configure discovery rules for that IP address then you may create a potential problem for Secret Server and the machines associated with that account.

Command Sets

Discovery command sets are customizable sets of commands that are sent over an SSH connection to the machines being accessed by Discovery.

When you are viewing your Discovery Sources on the Discovery Network View page there is a button to **Configure Command Sets**. Clicking on this takes you to a list of all of the custom command sets that are available to use with Discovery. You can select any existing command set to edit, or you can create a new one. When you create a new command set, you must give it a name and save it before you are able to enter commands.

In the **Command** and **Comment** grid, the left column is what will be sent to the machines during Discovery over an SSH communication, and the right column is just for commenting on the use or explaining the reason for the command.

The **Scan Type** dropdown has two options: **Find Machine** and **Find Local Accounts.**

- The **Find Machine** scan type is run during the Machine Discovery phase of Discovery
- The **Find Local Accounts** scan type is run during Local Account Discovery.

The commands in a command set will be executed sequentially in the order they are listed, but only the output of the final command will be returned.

ESX Discovery Source

Host Range

Host range settings will be automatically generated based on the **Find Machine** settings.

Find Machines

ESX/ESXi Discovery requires a list of individual IP addresses or DNS names to scan for accounts.

- **Lines** Each entry should be on its own line and must be a specific address rather than a range. Each line can have comma separated values in the following format if you want to provide additional data. IpAddress, Port, OperatingSystem, Hostname. Not all columns are required, but every entry needs to have a valid IP Address or DNS name at the very least.

Find Local Accounts

Define how you would like the ESX/ESXi Discovery Scanner to find local accounts on the ESX/ESXi machines.

- **Secret Credentials** Select the credentials to be used to scan with. These credentials may be Generic Discovery or VMware ESX/ESXi credentials. Multiple credentials may be used.
- **Port** The port that will be used to connect to the ESX or ESXi server in order to query for users.

Global settings

Navigate back through the **ADMIN** menu and select **Discovery** to return to the main Discovery page, then click **Edit**.

Select the **Enable Discovery** check box for the page to display the synchronization interval options.

The **Synchronization Interval** determines how often the regularly scheduled Discovery will run. If your network is particularly large or you have a very high number of computers and accounts you wish to discover, it is advisable to choose a greater period of time for the synchronization interval to ensure it has time to complete each time. Discovery can also be initiated manually at any time in addition to this regular schedule.

Discovery Configuration

─Discovery Settings─

ⓘ The AppPool running Secret Server must be configured to not shutdown. See the following KB Article.
Secret Server is currently running as "IIS APPPOOL\ss_qa".

Enable Discovery   ☑

Synchronization Interval for Discovery   Days   0
                                              Hours   1
                                              Minutes 0

Figure 24 Discovery Configuration Settings

Configuring Extensible Discovery

Extensible Discovery lets you extend the already powerful scanning abilities of Secret Server by creating custom scanners that run PowerShell. You can use one or more built-in or custom scanners at each step of the discovery process: host range discovery, machine discovery, local account discovery, and dependency discovery.

Extensible discovery revolves around two important concepts: Scan Templates and Discovery Scanners, also commonly referred to as Item Templates and Item Scanners, or just scanners. Standard Discovery also makes use of scan templates and discovery scanners, but are implemented out-of-the box and do not require any user configuration to work. If you want to use PowerShell scripts to perform any part of discovery you will need to define your own item templates and scanners and then add them to a new or existing discovery source.

WHY USE EXTENSIBLE DISCOVERY?

Creating a Discovery Source using scripted scanners can be a lot of work to set up, so when would you want to consider it? If you only need to find and import local administrator accounts as secrets and do not have any dependencies that you need to manage other than Windows Services, Application Pools, and Scheduled Tasks then you can use our existing built-in scanners and do not need to worry about extensible discovery. However, your network probably contains many other items that you want to be able to find and bring under managed control. Here are a just a few examples:

- Discover configuration files containing passwords and automatically add them as dependencies
- Scan computers not joined to the domain
- Create "dependencies" that run a SQL, SSH, or PowerShell script when the secret's password changes to log details about the password change event to an external source (e.g. an external auditing system or to raise an event on an external monitoring system).
- Bring information not currently imported by Local Account discovery back to custom fields in a Secret Template
- Discover SQL Server logins as "Local Accounts" and import them as SQL Server Account secrets

GETTING STARTED

When creating your first scripted Discovery Source, the easiest way to get started is through our new Extensible Discovery Configuration page. This page consolidates all of the areas that need to be configured into an easy-to-follow list. You can access this page through **ADMIN > Discovery** and clicking the **Extensible Discovery** button.

SCRIPTS

The first thing you will need to do when setting up any scripted Discovery Source is create the scripts that will be used by the discovery scanners. We currently only support PowerShell scripts for extensible discovery. SSH scripts are supported through the Command Sets feature. This will be explained further in the Discovery Sources section. SQL scripts are not currently supported for extensible discovery.

Go to the Scripts page by clicking the Scripts button on the Extensible Discovery Configuration page or by going to **ADMIN > Scripts** in the menu. Once there, click **Create New** in the PowerShell tab to create a new script. You can create scripts to discover Host Ranges (or OUs for Active Directory discovery sources), Machines, Local Accounts, and Dependencies.

Active Directory objects can be queried in PowerShell using Microsoft's Active Directory module for Windows PowerShell. Other systems can also be queried with different modules in order to discovery any resource you wish to discover. For examples of scripts that can be used with extensible discovery see the KB article referenced above.

PowerShell scripts used for discovery will need to return a list of objects that Secret Server can understand. These can be PowerShell objects that correspond to the types of objects discovered at each step of discovery such as the objects returned by Get-ADOrganizationalUnit and Get-ADComputer or they can be objects created using New-Object that have properties that can be interpreted by Secret Server. Secret Server interprets the output of discovery scripts through **Scan Templates**. Scan Templates are described in further detail below.

Discovery scripts can also be passed input through script parameters. This can be anything necessary to configure the script but is typically information such as domain names, ip addresses, computer names, and one or more credentials needed by the script. The **Discovery Scanner** section will discuss discovery script parameters in further detail.

Note   In order to run PowerShell scanners against machines for discovery of local accounts and dependencies you may need to configure WinRM and CredSSP. The following two KB articles describe how to do that:

- Configuring WinRM for PowerShell: http://ibm.biz/BdYBgF
- Configuring CredSSP for Use with WinRm/PowerShell: https://ibm.biz/BdYBgX
- WinRM Configuration for large-scale PowerShell RPC and Discovery: https://ibm.biz/BdYBgH

Script Category

Scripts have a new property: Category. You can use the "Filter by category" option to filter the list on the Scripts page for easier viewing. This may also be used by future updates to filter out inappropriate scripts in the various places that Secret Server where you can select a script (for example, custom password changers will only show "Heartbeat" scripts where you select what script to use to check for a successful heartbeat). If you upgraded to Secret Server 10.0 from an earlier version, any scripts that were created prior to upgrading will be

in the "Untyped" category. For backwards-compatibility, all places where a script can be selected will show any scripts in the appropriate category as well as any scripts in the "Untyped" category.

For scripts designed to be used in Extensible Discovery scanners, choose either the "Discovery Scanner" or "Untyped" category.

SCAN TEMPLATES

The goal of discovery scanning is to retrieve the following items:

- **Accounts** that can be imported and managed by as Secrets.
- Entities (**dependencies**) that need to be aware of password changes to managed Secrets.

The process of finding these two types of items usually involves the following scans prior to scanning for accounts and dependencies:

- **Host Ranges** where machines containing accounts can be found. For Active Directory, this usually involves scanning a domain for Organization Units or defining which Organization Units in a domain to check. For UNIX and ESXi this usually involved defining one or more lists of IP address ranges to scan.
- **Machines** to be scanned for accounts and dependencies.

Scan Templates define these objects. They are also used in several other places in Secret Server. Scan templates are the objects that link everything together in Discovery.

Figure 25 Extensible Discovery Configuration – Scan Templates

Secret Server defines scan templates for all of its built-in scanners and you should use these as input and output sources for your scripted scanners whenever possible. You will want to create your own scan templates

if you need to capture additional information as data for your scripts or if you need to use specific input and output templates on the Discovery Scanners in order to drive multiple discovery workflows on a single Discovery Source.

To create a new scan template, select the tab for the type of entity you want the template to represent and click **Create New Item Template**.

Figure 26 Extensible Discovery Configuration – Scan Template Designer

The scan template specifies at which step of the scanning process the item is created (Scan Type), the scan template from which the current template gets its required fields (Parent Scan Template), and the list of fields that the item will contain.

The Fields section describes the list of properties that will be returned by the built-in scanner or script for the item. At a minimum, you need to define one field for each of the fields on the parent scan template. For items that will be returned from a script you can also define additional fields that the script will return on the object. These will be mapped by name to the corresponding field on the scan template. These additional fields can then be used by later scanners.

Once you save a scan template the only changes you can make to it are changing the name and adding new fields. If you need to make any other changes you will need to create a new scan template, update anything that references the original template, and delete the original template. Built-in scanners cannot be deleted or edited except to add new fields.

DISCOVERY SCANNERS

Discovery Scanners define how a specific item – host range, machine, account, or dependency – is discovered. Item scanners can use either built-in methods or PowerShell scripts. Like scan templates, the "Discovery Scanners" page contains a tab for each scan step and built-in scanners for the default scan actions.

Figure 27 Extensible Discovery Configuration – Discovery Scanners

To add a scripted scanner, click **Create New Scanner**. When creating a new item scanner, you specify what type of scanner you are creating (Discovery Type), what type of Base Scanner to use (for example, Manual Input or Windows Discovery, or PowerShell Discovery), which Scan Item provides the input for the scan, and which Scan Item represents the output of the scan. If you use the "SSH Discovery" or "PowerShell Discovery" base scanner you will also be prompted to select a Command Set (optional) or Script and Script Arguments.

Figure 28 Extensible Discovery Configuration – New Discovery Scanner – SSH Discovery and PowerShell Discovery

Which scan templates you can use as Input and Output Templates is determined by the type of scanner you are creating.

- Host Range scanners accept the base Discovery Source as input and return entities that output to Host Range scan templates. You can select any of the built-in or custom scan templates of the "Find Host Ranges" scan type.
- Machine scanners accept Host Range scan templates as input and return entities that output to Machine scan templates.
- Local Account scanners accept Machine scan templates and return entities that output to Local Account scan templates.
- Dependency scanners also accept Machine scan templates and return entities that output to Dependency scan templates.

Since each scanner defines a specific input and output template you can create workflows through the discovery scanning process by matching the specific Output scan templates from one step to the Input scan templates at the next step. This will be described further in the next section.

Script Arguments

Script arguments can be a combination of literal values and tokens. When the script is run, these tokens are replaced with values from the input object and any privileged accounts associated with the scanner. Privileged accounts are assigned to item scanners when the item scanners are added to a discovery source.

The following table lists the tokens that can be used as script arguments:

Token	Description
$target	A generic placeholder for the input object. This is not used when scanning for Host Ranges because there is no previous scanner input source. For machine scanners, the $target refers to either the OU (for Active Directory Discovery Sources) or IP Address (for UNIX and ESXi Discovery Sources) from the Host Range input. For local account and dependency scanners the $target is the name of the computer being scanned.
$[x]$Username	The username of the xth privileged account associated with the scanner. Each scanner can have one or more privileged accounts associated with it. Thus, if you need to use the username of the first privileged account in your script you would pass in $[1]$Username. If you need to use the username of the second privileged account in your script you would pass in $[2]$Username, etc. You can have as many privileged accounts as necessary.
$[x]$Password	Like $[x]$Username, this is the password of the xth privileged account associated with the scanner.
$[x]$Domain	Like $[x]$Username and $[x]$Password, this is the fully-qualified domain name of the xth privileged account associated with the scanner.

DISCOVERY SOURCES

Discovery Sources create the workflow or series of workflows that a specific discovery scan will execute. Here you select which item scanners will be run at each step of the discovery process. As previously mentioned, the

Input and Output Templates define the flow of information through the discovery process. Each discovery step takes its input from the previous step, saves the items discovered at the current step, and sends those items to the appropriate scanner in the next discovery step. As a result, when you add Item Scanners to each step you can only select scanners that have an Input scan template matching the Output scan template of one of the scanners in the previous step. Also, no two item scanners in a step can have the same Output scan template.

When a scanner runs, it compares the results of the current scan with the results of the previous scan which are stored in the database. It updates any existing records with changes, adds new records for new items, and removes any records that do not match items found during the current scan. It does this based on the items' Scan Template. Thus, if there were more than one scanner with the same Output Template, the second scanner would end up removing the results of the first scan. This is why each Output Template must be unique.

To add a new scanner, click the appropriate **Add New [scanner type] Scanner** link. If there are any unassigned scanners that have an Input Template matching that of the previous step's Output Template and that have an Output Template type that is not used by any other assigned scanners, a dialog similar to the following one will be displayed. Otherwise, you will see a message letting you know that there are no valid scanners available. To select a scanner, click the "+" symbol next to the scanner's name.

Available Scanners

Select	Name	Base Scanner	Input Template	Output Template
⊕	Active Directory Local Admins	PowerShell Discovery	Windows Computer	Active Directory Account
⊕	Local Administrators	PowerShell Discovery	Windows Computer	Windows Local Admin
⊕	Local Non-Administrators	PowerShell Discovery	Windows Computer	Windows Local Non-Admins

✖ Cancel

Figure 29   Extensible Discovery Configuration – Available Scanners dialog

The "Find Host Ranges" step only allows a single item scanner. In all other steps you can add as many scanners as necessary. Thus, in the "Find Machines" step, you can only add items scanners whose Input scan

template match the Output scan template of the scanner added to the "Find Host Ranges" step. Since you can have multiple Machine scanners, each with its own Output scan template, the "Find Local Accounts" and "Find Dependencies" steps can have multiple item scanners with different Input scan templates.

You do not need to have an exact one-to-one correspondence between Item Scanners at each level. As you can see from the "Find Host Ranges" and "Find Machines" steps, one output can feed into multiple inputs. At the "Find Machine" and later levels you can create PowerShell Scanners that selectively process the items fed through their Input scan template. For example, one Machine scanner can process the output from the Host Range step to find one type of machine, another Machine scanner can process the same input to find a different type of machine, etc.

You could also have two Local Account Item Scanners defined that both take the same input from the previous "Find Machines" step. One of these scanners will examine each computer found in "Find Machines" for Windows local accounts while the other will examine each computer for AD accounts that have rights on the computer. Each will return its results to different Output Scan Templates, and those separate results can then be used to feed into Local Account Rules to create Windows Account secrets from the Windows local accounts and Active Directory Account secrets from the Active Directory accounts.

Running Discovery

The full Discovery process will run automatically corresponding to the synchronization interval specified in the Discovery configuration settings. However, if you wish to run Discovery manually, there are two steps, described below:

Run Discovery Manually

1. On the **Administration** menu, click **Discovery**.
2. Click **Run Now** above the **Discovery Log** (this runs the "Discovery Scan" - see below for more details).
3. Click **Run Now** above the **Computer Scan Log**, located toward the bottom of the page (this runs the "Computer Scan" – see below for more details).

DISCOVERY SCAN

When there is at least one active Discovery Source and Discovery is enabled, the ability to run a Discovery Scan will be made available. The button to **Run Now** will cause Discovery to begin running at your request, or you can wait for the interval settings to kick-in.

Active Directory Discovery

The Discovery Scan will use Active Directory calls to the specified domain(s) from the enabled Active Directory Discovery sources to find domain-joined machines on the network.

Unix Discovery

The Discovery Scan will use all of the specified hosts and host ranges from the Unix Discovery sources to find SSH-enabled machines on the network. If **Do Authentication** (enabled by default) is enabled for the Unix Discovery Sources, the selected command set for each Discovery Source will be executed to retrieve additional information about the computers that are found.

ESX Discovery

The Discovery scan will manually create an entry for the hosts specified in all VMware ESX/ESXi Discovery sources.

Discovery Log

When a Discovery scan runs, the output will appear in the Discovery log. This is where successes, failures, number of computers found, and number of rules applied will be noted.

COMPUTER SCAN

When there is at least one active Discovery Source, Discovery is enabled, and a Discovery Scan to find the machines on the network has run, a Computer Scan can be run with the **Run Now** button, or it will be automatically triggered by the Discovery interval.

Active Directory Discovery

When a computer scan runs, Active Directory queries will be used on each machine found during the Discovery scan to try and collect information regarding the types it has been enabled for, which can include: **Local Accounts**, **Windows Services**, **Scheduled Tasks**, and **IIS Application Pools**.

Unix Discovery

When a computer scan runs, the specified command set for each Unix Discovery Source will be run on the machines found during the Discovery scan to gather information for local users.

ESX Discovery

When a computer scan runs, it will attempt to find any users on the VMware ESX/ESXi machines that were specified.

Discovery Log

When a computer scan runs, the output will appear in the Computer Scan log. This is where successes, failures, number of local accounts found, and other information will be noted.

Discovery Network View

SEARCHING DISCOVERY RESULTS

To search for a specific Discovery Source or OU, type the source or OU name in the search bar displayed at left. If results are found, click the result shown below the search field to highlight it. Now, only machines from that source or OU will be displayed at right.

To search for a specific computer name, account, or service name, type the search term in the search field below the Local and Service Accounts tabs at right. Matching results will be filtered below the search field. To use advanced search settings, click **Advanced** beside the search field. Select an option in the **Search By** menu to narrow the search results to match an account, computer, or rule. **Account Status** can be configured to search for accounts that are unmanaged or managed by Secret Server. Selecting **Include Child Organization Units** will include matching search results within child OU's of the OU highlighted at the left.

Note Rule will only an advanced search option if Discovery Rules exist for local accounts. When this option is selected, another menu will appear to allow selection of a rule. For more information about creating and searching with rules, see Discovery Rules.

UNDERSTANDING DISCOVERY RESULTS

The table below describes the contents of each column, and which tabs the columns are present on.

Column	Description	Account Type (Local, Service)
Computer	Computer name of the machine scanned. This information is obtained from Active Directory during the first part of the Discovery process.	Both
Account	Username of the account discovered.	Both
Secret	If a Secret name appears in this column, a Secret already exists for the account listed in the Account column. Otherwise, this column will be blank.	Both
Last Scanned	Last date that the machine was scanned by Discovery.	Both
Status	Message indicating whether an account is managed by Secret Server as well as connectivity issues or lack of accounts detected. For more information about error messages, see Discovery Error Messages.	Both
Org Unit	Organizational Unit the machine is joined to. This information is obtained from Active Directory during the first part of the Discovery process.	Local
Last Connected	Last date any user logged into the machine.	Local
Service Name	Name of the dependency discovered.	Service

Type	Icon indicating the type of dependency discovered.	Service

Service account dependency types will be identified in the **Type** column as one of the following, and the **Icon** and **Service Name** columns will indicate additional details as follows:

Type	Icon	Service Name
Windows Service	⚙	Name of the service
Application Pool	📋	Name of the Application Pool in IIS
Scheduled Task	🕐	Name of the Scheduled Task

Note To correctly identify and import IIS Application Pools for IIS 7 or higher, Secret Server requires that there be a trust relationship between the scanned domain and domain that the Secret Server web server is joined to. For further details, see Application Pool Configuration for Discovery.

LOCAL ACCOUNTS

The **Local Account** tab will show all of the accounts that have been found for the Discovery Sources. The tree on the left allows you to filter which Discovery Source to view, as well as drilldown into OU's (Active Directory) or host ranges (Unix). The **Advanced** search allows filtering by account name, computer, operating system, or saved rules.
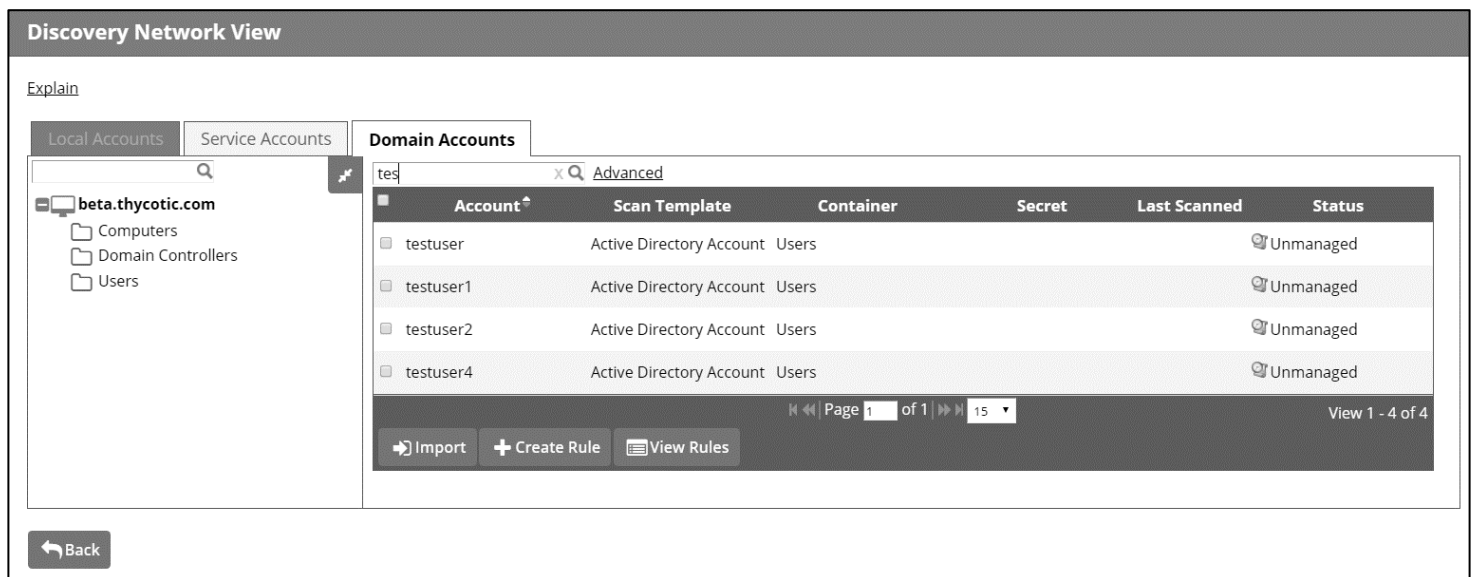
Figure 30 Local Accounts view

Import

The **Import** dialog wizard will guide you through importing an account as a Secret.

- **Secret** Select the Secret Type, Folder, and Secret Naming format for the Secret that will be created.
- **Password** If you know the current password for the selected accounts, leave that option selected, if you want Secret Server to change the password on the account as it creates the Secret, select the other option. This will allow you to choose a password for all new Secrets, or to have Secret Server randomly generate a new password for each new Secret created.
- **Import Password** If applicable, enter the new password for the Secrets to be created. Select the privileged Secret or Secrets that will be used to perform the initial password change on the account.

For Unix accounts, select the Password Type command set that will be used to take over the account. You can hold your cursor over the 'eye' icon to see the commands that will be used to change the password.

- **Password Changing** For Windows accounts choose whether this account will change its own password after the initial change, or if it will use a privileged account. For Unix rules, if prompted, decide which associated Secrets will be used as a part of the password change.
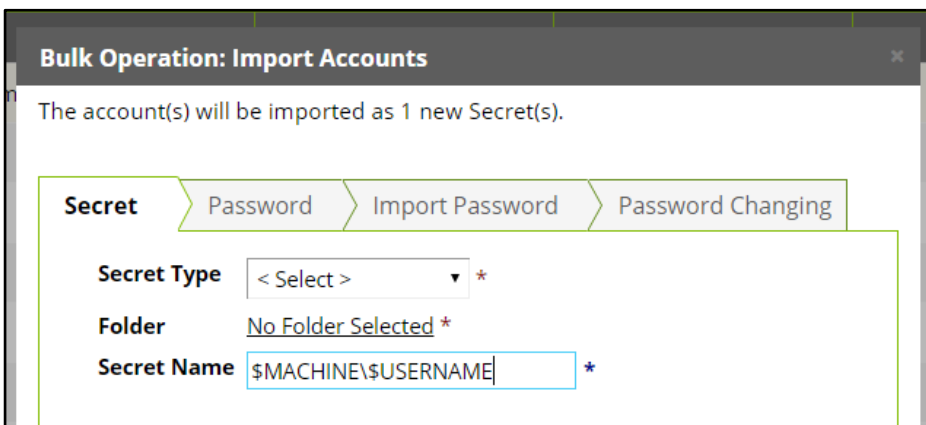
Figure 31 Import Accounts Wizard

Create Rule

The **Create Rule** dialog wizard will guide you through creating a rule that will can either create new Secrets as new accounts are discovered, send emails as new accounts are discovered, or both.

- **Rule** Enter the name and description for the new Rule
- **Source** Select the Discovery Source or sub folder for the rule to search on as well as how to filter which local accounts the rule will apply to. The first filter option is the Scan Template. For a standard discovery configuration without scripted scanners there should only be one option here. If you have added multiple local account scanners, then you will be able to select one of their Output scan templates. This will allow you to import the results of different scanners to different secret templates and with different settings. Rules can also filter by computer name, account name, and operating system name.

Note Pay attention to the OR/AND dropdown, as this can drastically change the results matching the rule.

- **Secret** If you want the rule to create new Secrets as account are discovered, check the Create Secrets option. Select the Secret Type, Folder, and Secret Naming format. The types of secrets available at this

step depend on the scan template selected in the previous step. This is determined by what password types are associated with the scan template. This association and how to configure it are described further in the **Password Types** section, below.

- **Password**   If you know the current password for accounts that the rule will match, leave that option selected, if you want Secret Server to change the password on the account as it creates the Secret, select the other option. This will allow you to choose a password for all new Secrets, or to have Secret Server randomly generate a new password for each new Secret created by the Rule
- **Import Password**   If applicable, enter the new password for the Secrets created by the rule. Select the privileged Secret or Secrets that will be used to perform the initial password change on the account.

For Unix Rules, select the Password Type command set that will be used to take over the account. You can hold your cursor over the 'eye' icon to see the commands that will be used to change the password.

- **Password Changing**   For Windows rules choose whether this account will change its own password after the initial change, or if it will use a privileged account. For Unix rules, if prompted, decide which associated Secrets will be used as a part of the password change.
- **Alerts**   If you want the rule to send an email when new accounts are found that match its search criteria, check the Send Email checkbox. Select users or groups that will receive the emails. You can also specify a **Take-Over** threshold. This will stop the import process if more accounts than are expected are found by Discovery.
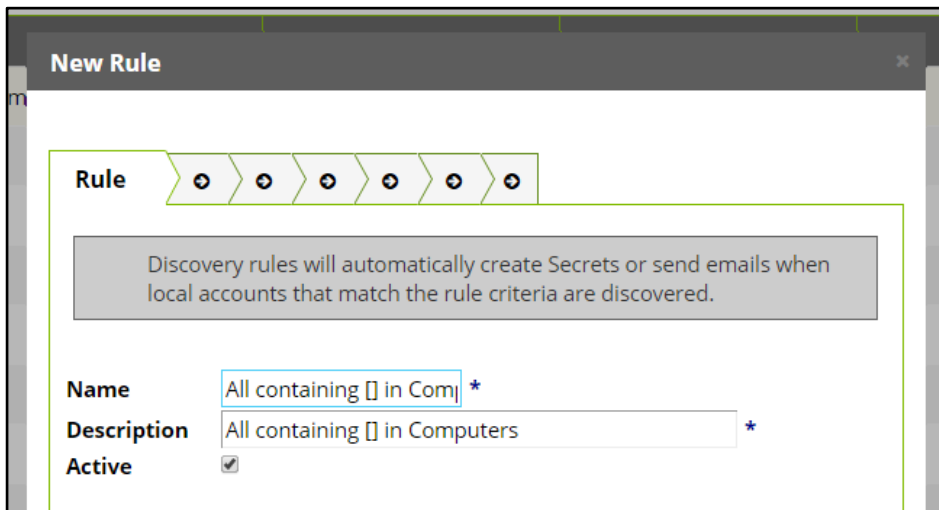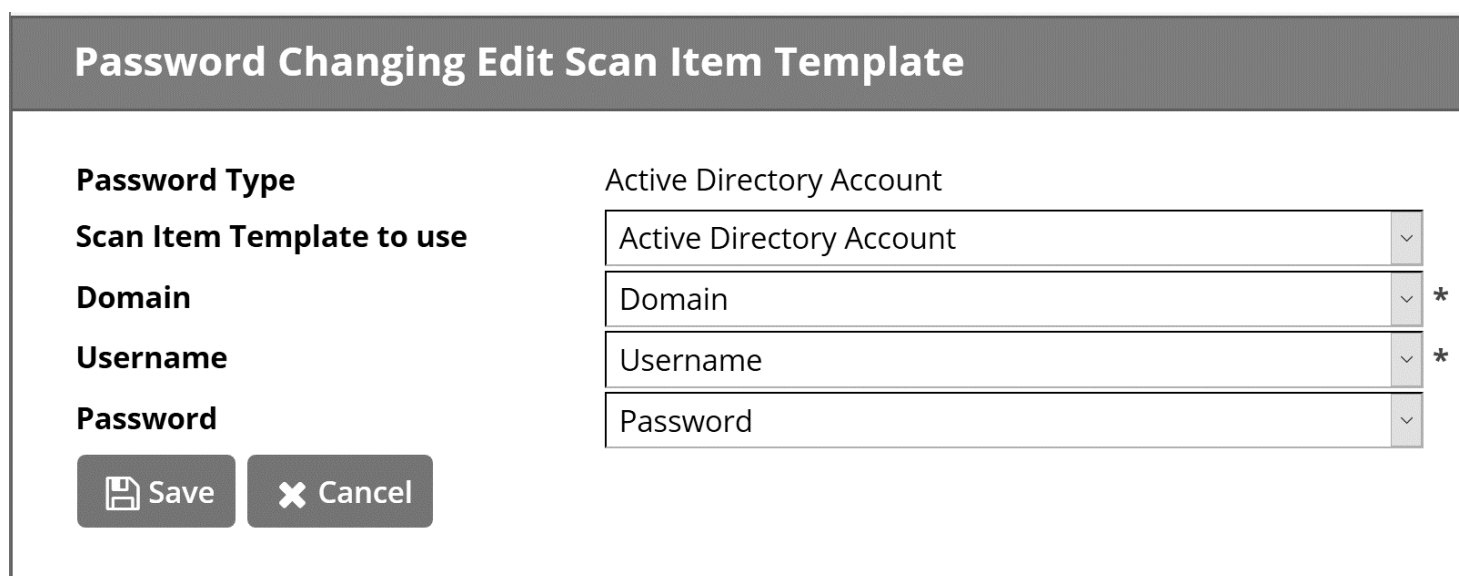

Figure 32   Create Rule Wizard

Note   Using a Discovery rule as a search filter will only apply to accounts that are found on computers in the OU's you are including in Discovery scan. To change those settings, modify the AD source to include more OU's or the entire domain.

Password Types

In addition to describing how to change and heartbeat a specific type of password, password types also handle the association between scan templates and secret types. Each password type can be associated with one

scan template. If a secret template allows password changing, it will have a password type associated with it. Multiple secret templates can use the same password type so one scan template can be associated with multiple secret templates. When you create a Local Account Rule and select the Scan Template to process by that rule, you will be able to assign all secrets created by that rule to one of the Secret Types that use any password type associated with that scan template.

Most of the built-in password types have already been assigned to scan templates. You can assign, unassign, or edit the scan template on any password type by selecting the type on the "Password Changers Configuration" page (**ADMIN > Remote Password Changing > Configure Password Changers**), clicking the **Configure Scan Template** button, then clicking **Edit**.

Password Changing Edit Scan Item Template

Password Type	Active Directory Account
Scan Item Template to use	Active Directory Account
Domain	Domain *
Username	Username *
Password	Password

💾 Save   ✖ Cancel
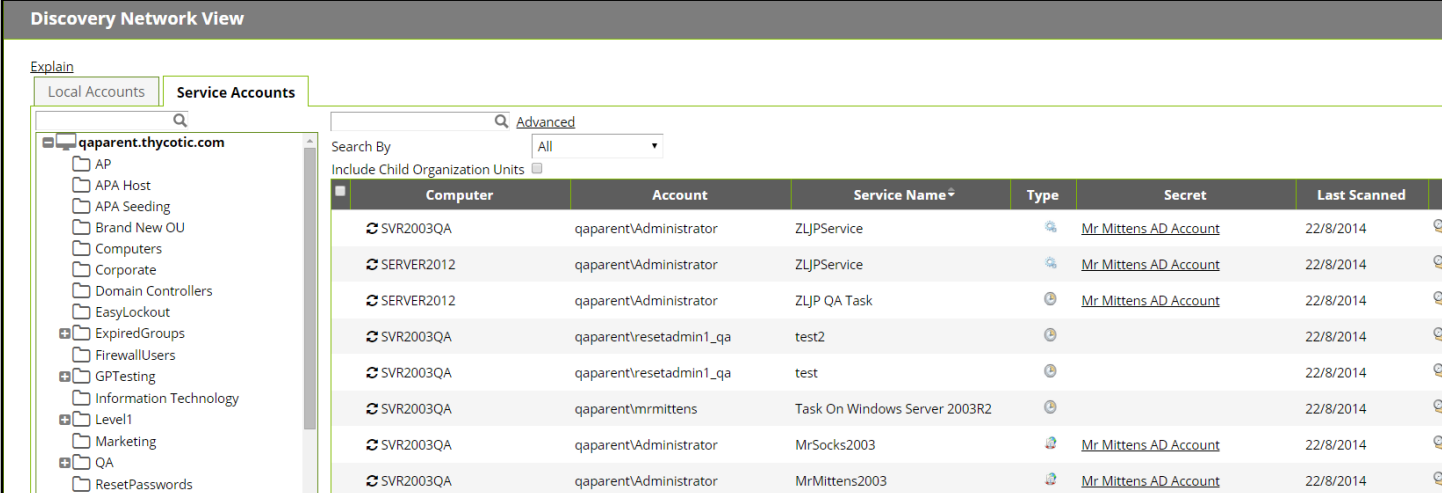
Figure 33   Password Changing Edit Scan Template

On the "Password Changing Edit Scan Template" page you can select which Local Account scan template to associate with the password type. When you select a scan template you will be presented with a list of fields from the template with a drop-down next to each. The values in the dropdown are the fields required by the password changer. Each of the values in the dropdown lists must be assigned to one of the fields from the scan template. If the scan template has additional fields that do not correspond to one of the required values for password changing, leave the dropdowns for those fields set to "<select>".

For example, if you are assigning a Scan Template to a Windows Local Account password type, you would need to assign the "machine," "username," and "password" fields from the password type to the corresponding fields on the Scan Template. When a Local Account Rule creates a secret it will then use this information to assign the fields from the Scan Template to the correct fields on the secret.

SERVICE ACCOUNTS

The Service Accounts Tab will show all of the places that Discovery found where an Active Directory account is being used to run a Windows Service, Scheduled Task, or IIS Application Pool. The **Advanced** search allows filtering by account name, computer, and service name.

Figure 34 Service Accounts view

Import

The **Import** option allows you to select several items in the list and import them as new Secrets with dependencies. If a Secret already exists for the account selected, a new dependency will be added to the existing Secret instead of creating a new one.

New Secrets

If this section is displayed, select the Secret Template, Folder, Secret Naming format for any new Secrets that will be created. Select whether the current password is known, or if Secret Server should change the password as the Secret is created. If a password change is desired, select a proper privileged Active Directory account that has permissions to reset the selected account's password.

New Dependencies

Optionally choose a privileged account that will perform the dependency update on all future password changes, and choose whether or not Secret Server will restart Windows Services after a password change for the new dependencies.
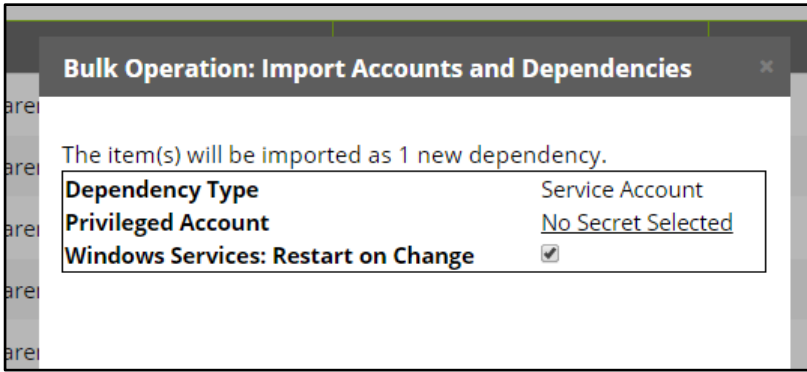
Figure 35 Import Dependency Wizard

Create Rule

Create a rule that will search within a domain or OU for new dependencies and as they are found and automatically add them to the matching existing Secrets as new dependencies. To create a dependency rule, select the discovery source or organizational unit to which the rule will apply. Next, select the Scan Template for the type of dependency that you want to import. For standard discovery, this will be either Windows Application Pool, Windows Scheduled Task, or Windows Service. If you have created additional dependency scan templates and assigned them to item scanners on the selected discovery source they will also show up here. The Dependency Template defines how the dependency will react when its associated secret's password is changed. The available templates will depend on the scan template selected. For a standard dependency configuration there should only be one option here. Dependency templates and how to create them are described in more detail in the **Dependencies** section of this document. You can also chose a secret to use as the privileged account that the dependency will use when its password change action is run.
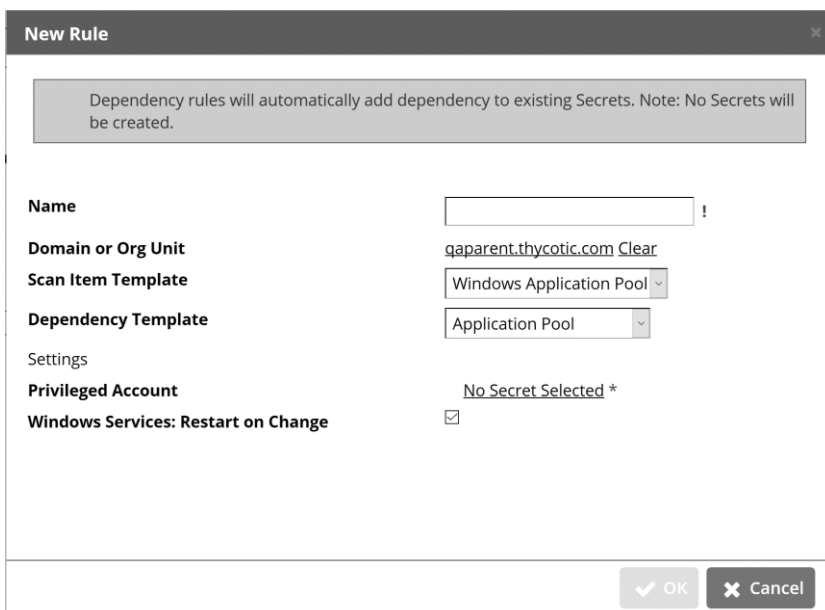
Figure 36 Dependency Rule Wizard

Note Using a Discovery rule as a search filter will <u>only</u> apply to accounts that are found on computers in the OU's you are including in Discovery scan. To change those settings, modify the AD source to include more OU's or the entire domain.

Dependencies

In order to set up effective dependency rules when using scripted discovery item scanners, you need to understand what happens when a password changes on a secret that has dependencies. When this occurs, an action is triggered for each dependency. These actions are defined by **Dependency Templates** and **Dependency Changers**. This action will update the dependency with the new password on the secret. Secret Server has built-in changers for the following types of dependencies:

- Application Pool
- Application Pool (recycle only)
- COM+ Application
- Remote File
- Scheduled Task
- Windows Service

Secret Server also supports scripted dependency changers. These can use PowerShell, SSH, or SQL scripts. In order to add PowerShell, SSH, or SQL dependencies you need to define dependency changers for them and associate those changers with dependency templates that can then be assigned to new dependencies. Dependencies can be added manually or through dependency rules, as we've seen above.

DEPENDENCY CHANGERS

Dependency Changers are the new way to define both a type of dependency and how to act on that dependency when the dependency secret's password changes. Each of the built-in dependency types has a pre-defined Dependency Changer. In addition, you can create additional dependency changers with customized settings for COM+, Remote File, PowerShell, SSH, and SQL dependency types. You can access the dependency changers page from **ADMIN > Discovery > Extensible Discovery > Configure Dependency Changers**.

Figure 37 Secret Dependency Changers

You must select a Scan Template for each custom dependency changer. This can be one of the pre-defined Dependency Scan Templates or a Dependency Scan Template that you have created. This determines the fields that are available to be set on manually-created dependencies. It also determines which dependency changer to assign to discovered dependencies that are added to secrets through Dependency Rules.

Each dependency changer also defines a "Wait (s)" value. This is an integer that specifies the number of seconds to wait before running the dependency. This is useful when a secret has multiple dependencies and you need to force a delay between the execution of each dependency's password changer.

Remote File Dependency Changers add the option to define a Regular Expression that specifies the placement of the password within a text file. This allows you to create Dependency Changers for different types of config files or connection strings in one place and reuse them on multiple dependencies. PowerShell Dependency scanners can also be created to scan machines for config files and -- using Dependency Rules -- automatically associate them with the matching secret and assign the correct Remote File Dependency Changer.

For more information about the Remote File dependency type and example regular expressions for several configuration files, refer to the following KB article: https://ibm.biz/BdYBgr

Figure 38 New Dependency Changer – Remote File

PowerShell Dependency Changers allow you to specify a PowerShell script to run as the password changer for the dependency. You must also specify what arguments to pass to the script when it is run. Script arguments can be literal values or translated argument tokens (see below for more information about translated arguments).

Figure 39 New Dependency Changer – PowerShell Script

SSH Dependency Changers allow you to specify a SSH script to run as the password changer for the dependency. You can also provide a port number if you need to override the default port. If the selected script has parameters, you will be prompted to provide values for each parameter. For literal parameters these will be literal values. For interpreted parameters these will be translated arguments tokens.

Figure 40 New Dependency Changer – SSH Script

SQL Dependency Changers allow you to specify a SQL script to run as the password changer for the dependency. Like SSH dependency changers, you can provide a port number if you need to override the default port. If the selected script has parameters, you will be prompted to provide values for each parameter. These values can be literal values or translated argument tokens. Finally, if the selected Script is based on a SQL type that requires an ODBC connection string, you will be prompted to provide values for the tokens in the ODBC connection string. Like script parameters, these can be literal values or translated argument tokens.

Figure 41 New Dependency Changer – SQL Script

Dependency Translated Argument Tokens

When entering values for script parameters or ODBC connection string arguments you can provide literal values or tokens that will be translated to a value defined on the dependency, its associated secret, or secrets

linked on the secret's "Remote Password Changing" tab. All of the tokens allowed prior to 10.0 are still valid plus the following new tokens. Tokens are not case-sensitive.

Token	Description
$SERVICENAME	The value of the Service Name field on the dependency. Service Name may have a different name based on the dependency type but is always the first part of the dependency title ("_____ on _____") as shown in the list of dependencies.
$MACHINE	The value of the Machine Name field on the dependency. This is always the second part of the dependency title ("_____ on _____") as shown in the list of dependencies.
${SCAN_ITEM_FIELD}	The name of any Scan Template field that is visible in the dependency edit dialog. If a scan item field is derived from a parent field, you may also use the parent field name as a parameter that translates to this field's value. For example, if you create a Scan Template and add a field called "ClientNetwork" based on the "Domain" parent field you can use either $CLIENTNETWORK or $DOMAIN and it will replace the token with the same value.

For a complete list of dependency tokens, refer to the following KB article: https://ibm.biz/BdYBga

DEPENDENCY TEMPLATES

Dependency Templates are the way the rest of the system interacts with dependency changers. In order to be used, a dependency changer must have a corresponding dependency template. The scan template on the dependency template must match the scan template on its associated dependency changer. When you create Dependency Rules or manually create dependencies you will specify the dependency changer by choosing a dependency template. You can access the dependency templates page from **ADMIN > Discovery > Extensible Discovery > Configure Dependency Templates**.

Figure 42 New Dependency Template dialog

SECRET DEPENDENCIES TAB

The addition of Dependency Templates and Changers has altered the way that dependencies are created or edited on a secret's Dependencies tab. Some of the information that you used to enter in the New/Edit Dependency dialog prior to version 10.0 is now entered on the dependency changer. For example: script, script parameters, and ODBC Connection Arguments are now entered on the dependency changer. However, the New/Edit Dependency dialog has added fields to enter values for the dependency's scan template fields (based on the Scan Template assigned to the Dependency Template selected for the dependency).

The exception to this is any PowerShell, SSH, or SQL dependency that existed prior to upgrading to 10.0. All other pre-existing dependencies will be converted during the upgrade to the corresponding pre-defined dependency template. PowerShell, SSH, and SQL dependencies, however, will be left as "legacy" dependencies on the secret. Legacy dependencies do not use a dependency template and still contain all the fields of pre-10.0 dependencies. You can edit all of the fields of a legacy dependency except the dependency type. If you want to replace a legacy template with one based on a dependency template, you will need to delete the legacy dependency and create a new dependency.

Figure 43 New Dependency dialog

To manually add a new dependency on a secret, click **Create New Dependency** from the secret's "Dependencies" tab. The "Templates" drop-down lists all dependency templates. If you select template that uses a script the dialog will display the script name for reference. You can hover over the eye icon to see the script in a preview window.

Depending on the dependency template, you can override some values from the underlying dependency changer when you add a new or edit an existing non-legacy dependency. For example, if you specify a value for "Wait (s)", "Port" (for SSH and SQL dependencies), or "Regex" (for Remote File dependencies) on the dependency changer it will appear as the default value on the dependency and you can provide an alternate value if necessary.