



# **Men & Mice Suite SOAP Command Reference**

## Table of Contents

<b>Introduction .....</b>	<b>7</b>
<b>General.....</b>	<b>7</b>
Authorization .....	7
Creating a SOAP session.....	7
Error handling.....	7
Referencing objects .....	8
Filtering.....	9
<b>Object types and properties .....</b>	<b>11</b>
Data Types.....	11
<i>DateTime</i> .....	11
<i>HexStr</i> .....	11
<i>IPAddress</i> .....	11
<i>IPOrHostname</i> .....	11
<i>ObjRef</i> .....	11
Enumerations .....	12
<i>AccessType</i> .....	12
<i>ADReplicationType</i> .....	12
<i>AuthenticationType</i> .....	12
<i>DHCPAddressPoolPermitType</i> .....	12
<i>DHCPLeaseField</i> .....	12
<i>DHCPReservationType</i> .....	12
<i>DHCPScopeOptionsReportType</i> .....	13
<i>DHCPServerSecurity</i> .....	13
<i>DHCPServerType</i> .....	13
<i>DNSRecordType</i> .....	13
<i>DNSServerType</i> .....	13
<i>DNSZoneType</i> .....	13
<i>EventType</i> .....	13
<i>HostDiscoveryType</i> .....	14
<i>ObjectType</i> .....	14
<i>PTRStatus</i> .....	14
<i>PropertyType</i> .....	14
<i>SortOrder</i> .....	14
<i>TimeUnit</i> .....	14
Objects .....	15
<i>AccessEntry</i> .....	15
<i>ADSite</i> .....	15
<i>AddressBlock</i> .....	15
<i>DHCPAddressPool</i> .....	15

<i>DHCPAddressPoolPermit</i> .....	16
<i>DHCPConcurrentLease</i> .....	16
<i>DHCPExclusion</i> .....	16
<i>DHCPGroup</i> .....	16
<i>DHCPLease</i> .....	17
<i>DHCPLeaseHistoryEntry</i> .....	17
<i>DHCPOption</i> .....	17
<i>DHCPOptionDefinition</i> .....	17
<i>DHCPReservation</i> .....	18
<i>DHCPScope</i> .....	18
<i>DCHPScopeOptionsReportEntry</i> .....	18
<i>DHCPServer</i> .....	19
<i>Discovery</i> .....	19
<i>DNSHost</i> .....	19
<i>DNSRecord</i> .....	20
<i>DNSServer</i> .....	20
<i>DNSView</i> .....	20
<i>DNSZone</i> .....	21
<i>Event</i> .....	21
<i>Folder</i> .....	21
<i>Group</i> .....	22
<i>HasAccessEntry</i> .....	22
<i>HostDiscoveryReportEntry</i> .....	22
<i>IdentityAccess</i> .....	23
<i>IPAMRecord</i> .....	23
<i>Object</i> .....	23
<i>ObjectAccess</i> .....	23
<i>ObjectError</i> .....	24
<i>Property</i> .....	24
<i>PropertyDefinition</i> .....	24
<i>Range</i> .....	25
<i>RangeWithStatistics</i> .....	25
<i>SystemSetting</i> .....	25
<i>User</i> .....	26
Arrays.....	27
<b>Command Reference</b> .....	<b>28</b>
Session specific commands.....	28
<i>Login</i> .....	28
<i>Logout</i> .....	28
Administration specific commands.....	29
<i>AddGroup</i> .....	29
<i>AddLicenseKey</i> .....	29
<i>AddUser</i> .....	29
<i>AddUserToGroup</i> .....	30
<i>GetGroup</i> .....	30
<i>GetGroups</i> .....	30

<i>GetHistory</i> .....	31
<i>GetSystemSettings</i> .....	31
<i>GetUser</i> .....	31
<i>GetUsers</i> .....	32
<i>RemoveUserFromGroup</i> .....	32
<i>SetPassword</i> .....	32
<i>SetSystemSettings</i> .....	33
DHCP specific commands .....	34
<i>AddDHCPAddressPool</i> .....	34
<i>AddDHCPExclusion</i> .....	34
<i>AddDHCPReservation</i> .....	34
<i>AddDHCPScope</i> .....	35
<i>AddDHCPServer</i> .....	35
<i>GetDHCPAddressPool</i> .....	35
<i>GetDHCPAddressPoolPermitList</i> .....	36
<i>GetDHCPAddressPools</i> .....	36
<i>GetDHCPBackupStatus</i> .....	36
<i>GetDHCPClassNames</i> .....	37
<i>GetDHCPExclusion</i> .....	37
<i>GetDHCPExclusions</i> .....	37
<i>GetDHCPGroup</i> .....	37
<i>GetDHCPGroups</i> .....	38
<i>GetDHCPLeaseHistory</i> .....	38
<i>GetDHCPLeases</i> .....	39
<i>GetDHCPOptionDefinitions</i> .....	39
<i>GetDHCPOptions</i> .....	39
<i>GetDHCPReservation</i> .....	40
<i>GetDHCPReservations</i> .....	40
<i>GetDHCPScope</i> .....	41
<i>GetDHCPScopes</i> .....	41
<i>GetDHCPServer</i> .....	41
<i>GetDHCPServers</i> .....	42
<i>GetDHCPSubclasses</i> .....	42
<i>ReleaseDHCPLease</i> .....	42
<i>SetDHCPOptions</i> .....	43
<i>SetDHCPSubclasses</i> .....	43
DNS specific commands .....	44
<i>AddDNSRecord</i> .....	44
<i>AddDNSRecords</i> .....	44
<i>AddDNSServer</i> .....	45
<i>AddDNSZone</i> .....	45
<i>GetADPartitions</i> .....	46
<i>GetADSite</i> .....	46
<i>GetADSites</i> .....	46
<i>GetDNSRecord</i> .....	47
<i>GetDNSRecords</i> .....	47
<i>GetDNSServer</i> .....	47

<i>GetDNSServers</i> .....	48
<i>GetDNSView</i> .....	48
<i>GetDNSViews</i> .....	48
<i>GetDNSZone</i> .....	49
<i>GetDNSZones</i> .....	49
<i>GetRelatedDNSRecords</i> .....	50
<i>PlaceInADSite</i> .....	50
<i>RemoveFromADSite</i> .....	50
IPAM specific commands.....	51
<i>AddRange</i> .....	51
<i>GetAddressBlocks</i> .....	51
<i>GetAddressesInUse</i> .....	51
<i>GetAvailableAddressBlocks</i> .....	52
<i>GetIPAMRecord</i> .....	52
<i>GetIPAMRecords</i> .....	53
<i>GetNextFreeAddress</i> .....	53
<i>GetRange</i> .....	53
<i>GetRanges</i> .....	54
<i>GetRangeStatistics</i> .....	54
<i>GetRangesWithStatistics</i> .....	55
<i>GetSubranges</i> .....	55
<i>GetRoutersForRange</i> .....	55
<i>JoinRanges</i> .....	56
<i>SetIPAMRecord</i> .....	56
<i>SetRoutersForRange</i> .....	57
<i>SplitRange</i> .....	57
Generic object management.....	58
<i>GetObjectAccess</i> .....	58
<i>GetObjects</i> .....	58
<i>GetProperties</i> .....	58
<i>GetPropertyDefinitions</i> .....	59
<i>HasAccess</i> .....	59
<i>RemoveObject</i> .....	59
<i>RemoveObjects</i> .....	60
<i>SetObjectAccess</i> .....	60
<i>SetProperties</i> .....	60
Miscellaneous commands.....	61
<i>AddFolder</i> .....	61
<i>CalcJoinedRanges</i> .....	61
<i>GetDHCPConcurrentLeases</i> .....	61
<i>GetFolder</i> .....	62
<i>GetFolderForObject</i> .....	62
<i>GetFolders</i> .....	62
<i>GetOrphanReverseDNSRecords</i> .....	63
<i>Ping</i> .....	63
<i>PlaceInFolder</i> .....	63
<i>ReportAccess</i> .....	63

<i>ReportDHCPScopeOptions</i> .....	64
<i>ReportHostDiscovery</i> .....	64
<i>ReportIPFromClientIdentifier</i> .....	65
<i>SendMail</i> .....	65
<b>Code Examples</b> .....	<b>66</b>
<b>Appendix – New and Changed Commands and Definitions</b> .....	<b>75</b>
New Data Types .....	75
New Commands .....	75
Changes to Data Types .....	75
Changes to Object Definitions .....	76
Changes to Existing Commands .....	76

## Introduction

This document contains information on the SOAP interface for the Men & Mice Suite. The SOAP interface was introduced in version 6.2 of the Men & Mice Suite. Version 6.2 contains several new commands and some commands and object types have been modified. A list of new commands and changes to existing commands and object types can be found in the Appendix.

**Important:** Version 6.3 of the Men & Mice Suite contains changes to both commands and object types. If you are using the SOAP interface for version 6.3, we strongly recommend that you view the Appendix for information on these changes.

## General

Many Men & Mice Central properties and methods are available through Simple Object Access Protocol (SOAP) commands. You can use the following URL to retrieve a list of supported SOAP services and WSDL (Web Services Definition Language):

### Windows:

[http://service\\_hostname:port/\\_mmwebext/mmwebext.dll?WSDL?server=mmcentral\\_server\\_hostname](http://service_hostname:port/_mmwebext/mmwebext.dll?WSDL?server=mmcentral_server_hostname)

### Unix:

[http://service\\_hostname:port/Soap?WSDL?server=mmcentral\\_server\\_hostname](http://service_hostname:port/Soap?WSDL?server=mmcentral_server_hostname)

- **service\_hostname** is the fully qualified domain name of the web server that is running the SOAP service.
- **mmcentral\_server\_hostname** is the fully qualified domain name of the server that is running Men & Mice Central.
- **port** is the port number to use for the web server that is running the SOAP service. The default port number is 80.

## Authorization

To run the SOAP commands, you must have the required access rights. You must be granted access to the Men & Mice Web interface to be able to use any of the SOAP commands.

## Creating a SOAP session

For an example on creating a SOAP session, see the **Code Examples** section.

## Error handling

If an error occurs when performing an action (for example if referring to a non-existing object or when trying to perform an action on an object without sufficient access privileges) an error message is returned. The error message uses the SOAP Fault message format. For further information see:

[http://www.w3schools.com/SOAP/soap\\_fault.asp](http://www.w3schools.com/SOAP/soap_fault.asp)

## Referencing objects

All objects in the Men & Mice Suite have a reference ID, **ref**, of type **ObjRef**. The reference is an ID that uniquely identifies an object in the Men & Mice Suite and is used to identify the object to be worked on.

A reference can either be a globally unique identifier (GUID), returned by the M&M Suite, or an object name. All functions in the SOAP interface try to resolve the reference intelligently, based on context. Functions working on more than one object type have an optional **ObjType** parameter that can be specified when an object is referenced by name, but which is not needed when a GUID is passed to them.

Since object names are not guaranteed to be unique, using GUIDs is the only way to guarantee that the M&M Suite will find the object. If more than one object have the same name, using that name as an object reference will cause an error.

### Example:

```
uint count = 0;
MMSOAP.Property[] props;

// This will fail because GetProperties() does not know the type of "my.zone."
props = mmSoap.GetProperties(mySession, "my.zone.", MMSOAP.ObjectType.Unknown, false,
false, false);

// This will fail if there are multiple zones with the name "my.zone."
props = mmSoap.GetProperties(mySession, "my.zone.", ObjectType.DNSZone, true, false,
false);

// This will select the first instance of "my.zone." to work on
MMSOAP.DNSZone[] zones = mmSoap.GetDNSZones(mySession, "name:my.zone.", 0, false, 0,
false, "name", MMSOAP.SortOrder.Ascending, true, out count);
props = mmSoap.GetProperties(mySession, zones[0].@ref, MMSOAP.ObjectType.Unknown, false,
false, false);
```

For object types where the "name" element can be non-unique, if a command is called giving an object reference, which is a non-unique name, an exception is thrown stating that the specified reference matches multiple objects.

But even for (some of) these object types, there is a way to reference objects with non-unique names:

- For zones, the reference can be in the format `[servername:]viewname:zonename`
- For scopes, the reference can be in the format `[servername:]scopename` or `[servername:]cidr`
- For views, the reference can be in the format `[servername:]viewname`

For example, to delete a zone called **test.com**. from the default view on server **dns1.com**:

```
<RemoveObject>
  <ref>dns1.com.::test.com.</ref>
```



```
<objType>DNSZone</objType>
</RemoveObject>
```

## Filtering

Some commands contain a **filter** element. This element can be used to filter results when getting a list of objects.

The filter has the following format:

```
[["]property["]:["][!][^]text[$]["][
["]property["]:["][!][^]text[$]["]]*
```

Where the "!" sign means does-not-match, the "^" sign means starts-with, and the "\$" sign means ends-with.

### Examples:

Filtering by string "mycorp" on all properties, you simply have:

```
<filter>mycorp</filter>
```

Filtering by string "mycorp" on property "name", you have:

```
<filter>name:mycorp</filter>
```

Filtering by string starting with "mycorp" on property "name":

```
<filter>name:^mycorp</filter>
```

Filtering by string not ending with "mycorp" on property "name":

```
<filter>name:!mycorp$</filter>
```

If the text must contain whitespace it must be quoted, so to filter by string not starting with "requested by" on property "comment":

```
<filter>comment:"!^requested by"</filter>
```

Same thing if the property name contains whitespace, so to filter by string "mycorp" on property "Company Name":

```
<filter>"Company Name":mycorp</filter>
```

When using a filter with GetDNSZones command, here's the filter to get all master zones, excluding reverse zones:

```
<filter>type:^Master$ name:!arpa.$</filter>
```

Getting only zones with name "domain.com.":

```
<filter>name:^domain.com.$</filter>
```

Getting only zones ending with name "domain.com.":

```
<filter>name:domain.com.$</filter>
```

Some examples that show the usage of a filter with the GetDNSRecords command. Remember that the command requires a dnsZoneRef element so you're always working relative to a specific zone. Here is the filter to get records having the name "time" from within the specified zone:

```
<filter>name:^time$</filter>
```

Getting all A records from within the specified zone:

```
<filter>type:^A$</filter>
```

Getting all records with data "ntp":

```
<filter>data:^ntp$</filter>
```

Getting all A records having the name "time":

```
<filter>type:^A$ name:^time$</filter>
```

## Object types and properties

This section contains a list of the object types and properties used by the Men & Mice Central SOAP interface.

### Data Types

The following data types are used by the Men & Mice Central SOAP interface.

---

#### **DateTime**

---

A string representing date and time.

Several date formats are supported. The date separator used determines the date format:

- If the first item is larger or equal to 1970 then the format is considered to be ymd
- If the date separator is '-' the format is considered to be [yy]yy-mm-dd
- If the date separator is '.' the format is considered to be dd.mm.yy[yy]
- If the date separator is '/' the format is considered to be mm/dd/yy[yy]

For time, a 24-hour clock is used. The format of the time should be:

- hour:minute[:Second[:msecond]]

If the time is omitted, the value of midnight (00:00) is used

Examples of supported DateTime formats:

- "2009-03-05 15:14" (Equals March 5, 2009, 3:14 PM)
- "05.03.09 15:14" (Equals March 5, 2009, 3:14 PM)
- "03/05/09 15:14" (Equals March 5, 2009, 3:14 PM)
- "2009-03-05" (Equals March 5, 2009, 00:00 AM)

---

#### **HexStr**

---

Hexadecimal string.

---

#### **IPAddress**

---

A valid IP address.

---

#### **IPOrHostname**

---

A valid IP address or a DNS host name.

---

#### **ObjRef**

---

A reference to an object.

## Enumerations

Various commands and objects use enumerations. Below is a list of the enumerations used by the Men & Mice Suite SOAP interface.

---

### AccessType

---

Defines possible access for an object property.

**Possible values:**

Unspecified, Allow, Deny

---

### ADReplicationType

---

Defines possible replication types for an AD integrated zone.

**Possible values:**

To\_All\_DNS\_Servers\_In\_AD\_Forrest, To\_All\_DNS\_Servers\_In\_AD\_Domain,  
To\_All\_Domain\_Controllers\_In\_AD\_Domain,  
To\_All\_Domain\_Controllers\_In\_Specified\_Partition, Unavailable

---

### AuthenticationType

---

Defines possible authentication types for a user that is logging in.

**Possible values:**

Internal, AD, ADGroup, RADIUS

---

### DHCPAddressPoolPermitType

---

Defines the types of DHCP Address Pool permits.

**Possible values:**

Unknown, AllClients, KnownClients, UnknownClients, DynamicBOOTPClients,  
MembersOfDHCPClass

---

### DHCPLeaseField

---

Defines the fields for a DHCP lease entry.

**Possible values:**

IPAddress, ClientIdentifier, ServerAddress, Hostname, Description

---

### DHCPReservationType

---

Defines possible DHCP reservation types.

**Possible values:**

DHCP, BOOTP, BOTH

---

### DHCPScopeOptionsReportType

---

Defines possible criteria for DHCP scope options report.

**Possible values:**

HasValue, DoesNotHaveValue, HasNoValueSet

---

### DHCPServerSecurity

---

Defines possible login methods for Cisco IOS DHCP servers.

**Possible values:**

Unknown, Telnet, SSH2

---

### DHCPServerType

---

Defines possible DHCP server types.

**Possible values:**

Unknown, MSDHCP, ISC, IOS

---

### DNSRecordType

---

Defines possible DNS record types.

**Possible values:**

Any, Unknown, A, NS, CNAME, SOA, WKS, PTR, HINFO, MX, TXT, SPF, RP, AFSDB, AAAA, LOC, SRV, NAPTR, DNAME, WINS, WINSR, SSHFP, DHCID, DS, RRSIG, DNSKEY, NSEC, NSEC3, NSEC3PARAM

---

### DNSServerType

---

Defines possible DHCP server types.

**Possible values:**

Unknown, MS, BIND

---

### DNSZoneType

---

Defines possible types for a DNS zone.

**Possible values:**

Master, Slave, Hint, Stub, Forward

---

### EventType

---

Defines possible event types.

**Possible values:**

Unknown, Creation, Modification, Deletion, Refresh, Error, Reference

---

### HostDiscoveryType

---

Defines possible host discovery types for host discovery report.

**Possible values:**

HostsWithData, HostsWithoutData

---

### ObjectType

---

Defines possible object types.

**Possible values:**

Unknown, Suite, DNSServer, DHCPSTerver, DNSZone, DNSView, Range, DHCPSTope, IPAddress, DHCPReservation, DHCPAddressPool, DHCPGroup, ADSite, DNSRecord, User, DHCPHost, Group, DHCPExclusion, DHCPFailoverPeer, DHCPLease, Folder

---

### PTRStatus

---

Defines possible status types for PTR records.

**Possible values:**

Unknown, OK, Verify

---

### PropertyType

---

Defines the possible data types for a custom property.

**Possible values:**

String, HexStr, Boolean, IPAddress, IPOrHostname, Integer

---

### SortOrder

---

Defines sorting order for a list.

**Possible values:**

Ascending, Descending

---

### TimeUnit

---

Defines possible time units.

**Possible values:**

Minutes, Hours, Days, Weeks, Months

---

## Objects

Objects contain actual object entries or references to the entries. Some objects may contain elements that are objects by themselves.

---

### AccessEntry

---

Contains the name and setting for an access entry relevant to an object.

#### Properties

Name	Type	Required	Nilable	Description
name	string	Yes	No	The name of the access entry.
access	AccessType	Yes	No	The setting for the access entry.

---

### ADSite

---

Contains properties for an AD Site object.

#### Properties

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	No	The name of the AD Site.
description	string	Yes	Yes	A description of the AD Site.
location	string	Yes	Yes	The location of the AD Site.

---

### AddressBlock

---

Contains all properties for an address block object.

#### Properties

Name	Type	Required	Nilable	Description
from	IPAddress	Yes	No	The start address of the address block.
to	IPAddress	Yes	No	The end address of the address block.

---

### DHCPAddressPool

---

Contains all properties for a DHCP address pool object.

#### Properties

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	No	No	The name of the DHCP address pool.
from	IPAddress	No	No	The start address of the address pool.
to	IPAddress	No	No	The end address of the address pool.
failoverPeer	string	No	No	The name of the failover peer if the address pool is on an ISC DHCP server with failover peer(s) defined.
dhcpScopeRef	ObjRef	Yes	No	A unique reference ID of the scope that contains the address pool.

---

**DHCPAddressPoolPermit**

---

Contains all properties for a DHCP address pool permit object.

**Properties**

Name	Type	Required	Nilable	Description
permit	DHCPAddressPoolPermitType	Yes	No	The DHCP address pool permit type
access	AccessType	No	No	The DHCP address pool access type
className	string	No	No	The name of the DHCP class if permit type is MembersOfDHCPClass

---

**DHCPConcurrentLease**

---

A lease entry for a lease that is active in more than one location.

**Properties**

Name	Type	Required	Nilable	Description
dhcpServerRef	ObjRef	Yes	No	A unique reference ID of the DHCP server that contains the lease.
dhcpServerName	string	Yes	No	The name of the DHCP server containing the lease.
dhcpScopeRef	ObjRef	Yes	No	A unique reference ID of the scope that contains the lease.
dhcpScopeName	string	Yes	No	The name of the scope that contains the lease.
name	string	Yes	Yes	The name of the lease.
type	string	Yes	Yes	The DHCP lease type.
mac	string	Yes	No	The MAC address of the host that holds the lease.
address	IPAddress	Yes	No	The IP address of the lease.
lease	string	Yes	No	The expiration date for the lease.
state	string	Yes	Yes	The state of the lease.

---

**DHCPExclusion**

---

Contains all properties for a DHCP exclusion object.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	No	No	The name of the exclusion.
from	IPAddress	No	No	The start address of the exclusion.
to	IPAddress	No	No	The end address of the exclusion.
dhcpScopeRef	ObjRef	Yes	No	A unique reference ID of the scope that contains the exclusion.

---

**DHCPGroup**

---

Contains all properties for a DHCP group object.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	No	No	The name of the DHCP group.
parentRef	ObjRef	Yes	No	An object reference to the parent group of the DHCP group. <b>null</b> if group has no parent group.



ownerRef      ObjRef      No      No      A unique reference ID of the DHCP group

### DHCPLease

Contains all properties for a DHCP lease object.

#### Properties

Name	Type	Required	Nillable	Description
name	string	Yes	Yes	The name of the lease.
type	string	No	No	The DHCP lease type.
mac	string	Yes	No	The MAC address of the host that holds the lease.
address	IPAddress	Yes	No	The IP address of the lease.
lease	string	Yes	No	The expiration date for the lease.
state	string	Yes	Yes	The state of the lease.
dhcpScopeRef	ObjRef	Yes	No	A unique reference ID of the scope that contains the lease.

### DHCPLeaseHistoryEntry

Contains all properties for a DHCP lease history entry object.

#### Properties

Name	Type	Required	Nillable	Description
serverAddress	string	Yes	No	The IP address of the DHCP server that issued the lease.
eventCode	string	Yes	Yes	A description of the event type for the lease entry.
timestampfirst	DateTime	Yes	Yes	The timestamp of the first history entry for the lease.
timestamplast	DateTime	No	Yes	The timestamp of the last history entry for the lease.
ipAddress	string	Yes	Yes	The IP address of the lease.
clientidentifier	string	Yes	Yes	The identifier of the client.
hostname	string	Yes	Yes	The host name for the lease entry.
numberOfOccurrences	string	No	Yes	The number of history entries we have for the lease.

### DHCPOption

Contains a name and a value for a DHCP option.

#### Properties

Name	Type	Required	Nillable	Description
option	string	Yes	No	The DHCP option ID.
value	string	No	Yes	The DHCP option value.

### DHCPOptionDefinition

Contains a definition for a DHCP option.

#### Properties

Name	Type	Required	Nillable	Description
option	string	Yes	No	The DHCP option ID.
name	string	Yes	No	The DHCP option name.
type	PropertyType	Yes	No	The DHCP option type.

---

**DHCPReservation**

---

Contains all properties for a DHCP reservation object.

**Properties**

Name	Type	Required	Nullable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	No	The name of the reservation.
clientIdentifier	string	No	Yes	The client identifier used for the reservation (e.g. the MAC address).
addresses	ArrayOfIPAddress	No	Yes	An array of zero or more IP addresses used for the reservation.
type	DHCPReservationType	No	No	The type of the DHCP reservation
description	string	No	Yes	Description for the reservation. Only applicable for MS DHCP servers.
ddnsHostName	string	No	Yes	Dynamic DNS host name for reservation. Only applicable for ISC DHCP servers.
filename	string	No	Yes	The "filename" DHCP option. Only applicable for ISC DHCP servers.
serverName	string	No	Yes	The "server-name" DHCP option. Only applicable for ISC DHCP servers.
nextServer	string	No	Yes	The "next-server" ISC DHCP option. Only applicable for ISC DHCP servers.
ownerRef	ObjRef	Yes	No	A unique reference ID of the DHCP server, group or scope that contains the reservation.

---

**DHCPScope**

---

Contains all properties for a DHCP scope object.

**Properties**

Name	Type	Required	Nullable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	No	The name of the DHCP scope.
rangeRef	ObjRef	Yes	No	A reference to the range associated with the scope.
dhcpServerRef	ObjRef	Yes	No	A reference to the DHCP server containing the scope.
superscope	string	No	Yes	The name of the superscope for the DHCP scope. Only applicable for MS DHCP servers.
description	string	No	Yes	A description for the DHCP scope.
available	unsignedInt	No	No	Numer of available addresses in the address pool(s) of the scope.
enabled	Boolean	No	No	Specifies whether the scope is enabled (default <b>true</b> ).

---

**DCHPScopeOptionsReportEntry**

---

Contains an entry for the DHCP Scope Options report.

**Properties**

Name	Type	Required	Nullable	Description
dhcpServerName	string	Yes	No	The name of the DHCP server.
dhcpScopeName	string	Yes	No	The name of the DHCP scope.
optionValue	string	Yes	Yes	The value of the DHCP option.
inherited	Boolean	Yes	No	Specifies whether the option is inherited.

---

**DHCP**Server

---

Contains all properties for a DHCP server object.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	No	The name of the DHCP server.
proxy	string	No	No	The DHCP server controller used to connect to the DHCP server.
address	IPAddress	No	No	The IP address of the DHCP server.
resolvedAddress	IPAddress	No	No	The IP address of the DHCP server, if specified, or the IP address retrieved through resolving the name of the DHCP server.
username	string	No	Yes	The login name of the user accessing a Cisco IOS DHCP server. Only applicable for Cisco IOS DHCP servers.
password	string	No	Yes	The password for the user accessing a Cisco IOS DHCP server. Only applicable for Cisco IOS DHCP servers.
enablePassword	String	No	Yes	The password for entering maintenance mode when accessing a Cisco IOS DHCP server. Only applicable for Cisco IOS DHCP servers.
type	DHCPServerType	Yes	No	The DHCP server type.
security	DHCPServerSecurity	No	No	The login method used for the DHCP server. Only applicable for Cisco IOS DHCP servers.
customProperties	ArrayOfProperty	No	Yes	The custom properties for the DHCP server.

---

**Discovery**

---

Contains all properties for a ping discovery schedule that is used for ranges or scopes.

**Properties**

Name	Type	Required	Nilable	Description
interval	unsignedInt	Yes	No	The interval between runs for the discovery schedule.
unit	TimeUnit	Yes	No	The time unit for the interval.

---

**DNS**Host

---

Contains all properties for a DNS host object. A DNS host is an address (A) record and all CNAME, TXT and MX records that are associated with the record.

**Properties**

Name	Type	Required	Nilable	Description
dnsRecord	DNSRecord	Yes	No	The DNS host record. The record must be an address (A) record. Note that the name element of the record is fully qualified here.
ptrStatus	PTRStatus	No	No	The PTR record status for the host record.
relatedRecords	ArrayOfDNSRecord	Yes	Yes	An array of records associated with the host record.

---

**DNSRecord**

---

Contains all properties for a DNS record object.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	Yes	The name of the DNS record, either partially or fully qualified.
type	DNSRecordType	Yes		Specifies the type of the DNS record.
ttl	string	Yes	Yes	Specifies the Time-To-Live of the DNS record.
data	string	Yes	Yes	Contains the record data in a tab-separated list.
comment	string	No	Yes	Contains the comment string for the record. Note that only records in static DNS zones can have a comment string.
enabled	Boolean	Yes	No	Contains <b>true</b> if the record is enabled. If the record is disabled the value is <b>false</b> .
aging	unsignedInt	No	No	The aging timestamp of dynamic records in AD integrated zones.
dnsZoneRef	ObjRef	Yes	No	Contains the reference ID of the zone that contains the record.

---

**DNSServer**

---

Contains all properties for a DNS server object.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	No	The name of the DNS server.
address	IPAddress	No	No	The IP address of the DNS server.
resolvedAddress	IPAddress	No	No	The IP address of the DNS server, if specified, or the IP address retrieved through resolving the name of the DNS server.
port	unsignedInt	No	No	The port to use when connecting to the DNS server. Only needed if a non-standard port is used to connect to the server.
proxy	string	No	No	The DNS Server Controller used to connect to the DNS server.
type	DNSServerType	No	No	The type of DNS server.
adForest	String	No	No	The name of the AD Forest which the DNS server is associated with.
customProperties	ArrayOfProperty	No	Yes	The custom properties for the DNS server.

---

**DNSView**

---

Contains information about a DNS view.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	Yes	No	A reference to the object.
name	string	Yes	Yes	The name of the DNS view.
dnsServerRef	ObjRef	Yes	No	A reference to the server containing the view.

---

**DNSZone**

---

Contains all properties for a DNS zone object.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	No	The name of the zone.
dynamic	boolean	No	No	Contains <b>true</b> if the zone is dynamic.
adIntegrated	boolean	No	No	Contains <b>true</b> if the zone is Active Directory integrated.
adReplicationType	ADReplicationType	No	No	The AD replication type if the zone is Active Directory integrated.
adPartition	String	No	Yes	The AD partition if the zone is Active Directory integrated.
dnsViewRef	ObjRef	No	No	Contains the reference ID of the DNS view that contains the zone. This property is not supplied for AD integrated zones.
dnsViewRefs	ArrayOfObjRef	No	No	Contains the reference IDs of the DNS views that contain the zone.
authority	string	No	No	Contains the name of the DNS server that contains the zone or the string "[Active Directory]" if the zone is AD integrated.
type	DNSZoneType	Yes	No	Specifies the type of the zone.
customProperties	ArrayOfProperty	No	Yes	The custom properties for the zone.

---

**Event**

---

Contains information about an event.

**Properties**

Name	Type	Required	Nilable	Description
eventType	EventType	Yes	No	The type of event.
objType	ObjectType	Yes	No	The type of object associated with the event.
objRef	ObjRef	Yes	No	A reference to the object associated with the event.
objName	string	Yes	Yes	The name of the object.
timestamp	DateTime	Yes	No	The time when the event occurred.
username	string	Yes	No	The login name of the user initiating the event.
saveComment	string	Yes	Yes	The save comment for the event.
eventText	string	Yes	Yes	The event description.

---

**Folder**

---

Contains all properties for a folder object.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	No	The name of the group.
contentType	ObjectType	Yes	No	The type of objects contained within the folder.
parentRef	ObjRef	No	No	A reference to the parent folder.

---

## Group

---

Contains all properties for a user group object.

### Properties

Name	Type	Required	Nilable	Description
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	No	The name of the group.
description	string	No	Yes	A description for the group.
adIntegrated	boolean	No	No	Contains <b>true</b> if the group is Active Directory integrated.
groupMembers	ArrayOfObject	No	Yes	An array of references to the users in the group.

---

## HasAccessEntry

---

An entry that contains information about whether the currently logged in user has access for a specific access entry for a specific object.

### Properties

Name	Type	Required	Nilable	Description
ref	ObjRef	Yes	No	A unique reference ID for the object.
objType	ObjectType	Yes	No	The type of object.
name	string	Yes	No	The name of the object.
hasAccess	boolean	Yes	No	Contains <b>true</b> if the user has access for the access entry.

---

## HostDiscoveryReportEntry

---

An entry that contains information for discovered host. Used by some of the reports in the system.

### Properties

Name	Type	Required	Nilable	Description
addrRef	ObjRef	Yes	No	A unique reference ID for the IP address of the host.
address	string	Yes	No	The IP address of the host.
dnsHostNames	string	Yes	Yes	A comma-separated list of DNS host names associated with the IP address.
lastSeenDate	DateTime	Yes	No	The date when the IP address last responded to a ping request.
lastDiscoveryDate	DateTime	Yes	No	The date when the system last checked whether the IP address responded to a ping request.
customProperties	ArrayOfProperty	Yes	Yes	The custom properties for the IP address.

---

**IdentityAccess**

---

Contains access entries for a specific user or group.

**Properties**

Name	Type	Required	Nilable	Description
identityRef	ObjRef	Yes	No	A unique reference ID for a user or group for which the access applies.
identityName	string	No	No	The name of the user or group for which the access applies.
accessEntries	ArrayOfAccessEntry	Yes	Yes	The access entries for the user or group.

---

**IPAMRecord**

---

Contains all properties for an IPAM record object.

**Properties**

Name	Type	Required	Nilable	Description
addrRef	ObjRef	Yes	No	A reference to the object.
address	string	Yes	No	IP address of the record.
claimed	boolean	No	No	Contains <b>true</b> if the record has the Claimed flag set.
dnsHosts	ArrayOfDNSHost	Yes	Yes	List of DNS hosts associated with the address.
dhcpReservations	ArrayOfDHCPReservation	Yes	Yes	List of DNS reservations associated with the address.
dhcpLeases	ArrayOfDHCPLease	Yes	Yes	List of DHCP leases associated with the address.
lastSeenDate	string	Yes	Yes	The date when the address last responded to a ping request.
lastDiscoveryDate	string	Yes	Yes	The date when the system last checked whether the address responded to a ping request.
ptrStatus	PTRStatus	No	No	The PTR record status for the record.
extraneousPTR	Boolean	No	No	Contains <b>true</b> if there are extraneous PTR records for the record.
customProperties	ArrayOfProperty	Yes	Yes	The custom properties for the IPAM record.

---

**Object**

---

Contains information about a generic object.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	Yes	No	A reference to the object.
objType	ObjectType	No	No	The type of the object.
name	string	No	No	The name of the object.

---

**ObjectAccess**

---

Contains information about an object access entry.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	Yes	No	A reference to the object.
name	string	Yes	No	The name of the object.
identityAccess	ArrayOfIdentityAccess	Yes	No	An array of access entries for different users and groups for the object.

**ObjectError**

Contains information about an error related to an operation on an object.

**Properties**

Name	Type	Required	Nilable	Description
ref	ObjRef	Yes	No	A reference to the object.
error	string	Yes	No	The error string.

**Property**

Contains a single property entry for an object.

**Properties**

Name	Type	Required	Nilable	Description
name	string	Yes	No	The name of the property.
value	string	No	Yes	The value of the property.

**PropertyDefinition**

Contains the property definition for a property.

**Properties**

Name	Type	Required	Nilable	Description
name	string	Yes	No	The name of the property.
type	PropertyType	No	No	The type of the property.
system	boolean	No	No	Contains <b>true</b> if the property is system defined.
mandatory	boolean	No	No	Determines if the property is mandatory.
readOnly	boolean	No	No	Determines if the property is read-only.
multiLine	boolean	No	No	Determines if the property is multi-line.
defaultValue	string	No	Yes	Contains the default value for the property.
listItems	ArrayOfString	No	Yes	Contains the list items if the property is an enumeration.



---

## Range

---

Contains all properties for a range object.

### Properties

Name	Type	Required	Nullable	Description
ref	ObjRef	No	No	A reference to the object.
name	string	Yes	No	The name of the range.
from	IPAddress	No	No	The starting IP address of the range.
to	IPAddress	No	No	The ending IP address of the range
parentRef	ObjRef	No	No	A reference to the immediate parent range.
adSiteRef	ObjRef	No	No	A reference to the AD site to which the range belongs.
childRanges	ArrayOfObject	No	Yes	An array of child ranges of the range.
dhcpScopes	ArrayOfObject	No	Yes	An array of scopes associated with the range.
subnet	boolean	No	No	Determines if the range is defined as a subnet.
locked	boolean	No	No	Determines if the range should be set as locked.
autoAssign	boolean	No	No	Determines if it should be possible to automatically assign IP addresses from the range.
hasSchedule	boolean	No	No	Determines if a discovery schedule has been set for the range.
customProperties	ArrayOfProperty	No	Yes	The custom properties for the range.

---

## RangeWithStatistics

---

Contains all information about a range along with statistic information for the range.

### Properties

Name	Type	Required	Nullable	Description
range	Range	Yes	No	The range.
used	string	Yes	No	The number of used IP addresses in the range.
free	string	Yes	No	The number of free IP addresses in the range.
utilization	unsignedInt	Yes	No	The utilization percentage for the range.

---

## SystemSetting

---

Contains properties for a SystemSetting object.

### Properties

Name	Type	Required	Nullable	Description
name	string	Yes	No	Name of the system setting.
value	string	No	No	Value of the system setting.

---

**User**

---

Contains all properties for a user object.

**Properties**

<b>Name</b>	<b>Type</b>	<b>Required</b>	<b>Nullable</b>	<b>Description</b>
ref	ObjRef	No	No	A unique reference ID for the object.
name	string	Yes	No	The login name of the user.
password	string	No	Yes	The user password.
fullName	string	No	Yes	The full name of the user.
description	string	No	Yes	The description for the user.
authenticationType	AuthenticationType	Yes	No	The authentication type for the user.
groups	ArrayOfObject	No	Yes	An array of the groups the user is a member of.

## Arrays

Arrays exist for most object types. Arrays contain zero or more entries of an object of a certain type. Arrays are used for many commands, both as input parameters and return parameters. Below is an example for the properties of `ArrayOfDNSServer`:

<b>Name</b>	<b>Type</b>	<b>Occurrences</b>	<b>Description</b>
dnsServer	DNSServer	0-n	Entry of type DNSServer. The array can contain 0-n entries of this type.

## Command Reference

### Session specific commands

The following commands are used for session management.

---

#### Login

---

Logs into the Men & Mice Suite through the SOAP interface

##### Parameters

Name	Type	Required	Nillable	Description
server	string	Yes	No	DNS name or IP address of the server running Men & Mice Central.
loginName	string	Yes	No	Login name of user.
password	string	Yes	No	User password.
replaceControlCharacters	Boolean	No	No	Determines if invalid control characters in user data should be replaced by a question mark.

##### Return value

Name	Type	Description
session	string	Session ID for this login.

---

#### Logout

---

Logs out of the Men & Mice Suite.

##### Parameters

Name	Type	Required	Nillable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.

##### Return value

This command has no return value.

## Administration specific commands

The following commands are used for various administrative tasks.

---

### AddGroup

---

Adds a new user group.

#### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
group	Group	Yes	No	The group to add. See description of Group in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.

#### Return value

Name	Type	Description
ref	ObjRef	Reference to the created record. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

### AddLicenseKey

---

Adds a new license key to the system.

#### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
licenseKey	string	Yes	No	The license key to add.

#### Return value

This command has no return value.

---

### AddUser

---

Adds a new user.

#### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
user	User	Yes	No	The user to add. See description of User in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.

#### Return value

Name	Type	Description
ref	ObjRef	Reference to the created record. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

## AddUserToGroup

---

Adds a user to a group.

### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
userRef	ObjRef	Yes	No	A reference to the user to add.
groupRef	ObjRef	Yes	No	A reference to the group to which the user should be added.
saveComment	string	No	Yes	Save comment text to associate with the action.

### Return value

Name	Type	Description
ref	ObjRef	Reference to the created record. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

## GetGroup

---

Gets a group object based on the object reference.

### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
groupRef	ObjRef	Yes	No	An object reference for the group.

### Return value

Name	Type	Description
group	Group	The group object.

---

## GetGroups

---

Gets a list of user groups.

### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

### Return value

Name	Type	Description
groups	ArrayOfGroup	List of groups. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

## GetHistory

---

Gets a list of history entries.

### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
eventType	EventType	No	No	The type of event to retrieve. If not specified, entries for all event types are retrieved.
objType	ObjectType	No	No	The type of object to retrieve. If not specified, entries for all object types are retrieved.
objRef	ObjRef	No	No	If specified, only history entries for the specified object will be retrieved.
username	string	No	Yes	The login name of the user for which to retrieve history entries. If not specified, entries for all users are retrieved.
startDate	DateTime	No	Yes	Date of the first entry to retrieve.
endDate	DateTime	No	Yes	Date of the last entry to retrieve.
filter	string	No	Yes	Filtering criteria for the records returned.
limit	unsignedInt	No	No	The number of records to return.

### Return value

Name	Type	Description
events	ArrayOfEvent	List of events. See description of arrays the <b>Object Types and Properties</b> section.

---

## GetSystemSettings

---

Gets the system settings for the Men & Mice Suite.

### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.

### Return value

Name	Type	Description
systemSettings	ArrayOfSystemSetting	List of system settings. See description of arrays the <b>Object Types and Properties</b> section.

---

## GetUser

---

Gets a user object based on the object reference.

### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
userRef	ObjRef	Yes	No	An object reference for the user.

### Return value

Name	Type	Description
user	User	The user object.

---

## GetUsers

---

Gets a list of users.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

### Return value

Name	Type	Description
users	ArrayOfUser	Array of users. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

## RemoveUserFromGroup

---

Removes a user from a group.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
userRef	ObjRef	Yes	No	A reference to the user to remove.
groupRef	ObjRef	Yes	No	A reference to the group from which the user should be removed.
saveComment	string	No	Yes	Save comment text to associate with the action.

### Return value

This command has no return value.

---

## SetPassword

---

Changes the password for the currently logged in user.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
oldPassword	string	Yes	No	The old password for the user.
newPassword	string	Yes	No	The new password for the user.

### Return value

This command has no return value.



---

## SetSystemSettings

---

Changes the system settings for the Men & Mice Suite.

### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
systemSettings	systemSettings	Yes	Yes	The system settings to set.

### Return value

This command has no return value.

## DHCP specific commands

The following commands are used for managing DHCP data. Note that various commands in the generic object management section are used to manage DHCP data as well. See the section **Generic object management** for more information on these commands.

---

### AddDHCPAddressPool

---

Adds a new DHCP address pool to a scope. Note that a scope on an MS DHCP server cannot contain more than one DHCP address pool.

#### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpAddressPool	DHCPAddressPool	Yes	No	The DHCP address pool to add. See description of DHCPAddressPool in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.

#### Return value

Name	Type	Description
ref	ObjRef	Reference to the new DHCP address pool. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

### AddDHCPExclusion

---

Adds a new DHCP exclusion to a scope.

#### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpExclusion	DHCPExclusion	Yes	No	The DHCP exclusion to add. See description of DHCPExclusion in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.

#### Return value

Name	Type	Description
ref	ObjRef	Reference to the new DHCP exclusion. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

### AddDHCPReservation

---

Adds a new DHCP reservation to a scope.

#### Parameters

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpReservation	DHCPReservation	Yes	No	The DHCP reservation to add. See description of DHCPReservation in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

Name	Type	Description
ref	ObjRef	Reference to the new DHCP reservation. See description of ObjRef in the <b>Object Types and Properties</b> section.

**AddDHCPscope**

Creates a new DHCP scope.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpScope	DHCPscope	Yes	No	The DHCP scope to add. See description of DHCPscope in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

Name	Type	Description
ref	ObjRef	Reference to the new DHCP scope. See description of ObjRef in the <b>Object Types and Properties</b> section.

**AddDHCPserver**

Adds a DHCP server.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpServer	DHCPserver	Yes	No	The DHCP server to add. See description of DHCPserver in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

Name	Type	Description
ref	ObjRef	Reference to the new DHCP server. See description of ObjRef in the <b>Object Types and Properties</b> section.

**GetDHCPAddressPool**

Gets a DHCP address pool object based on the object reference.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpAddressPoolRef	ObjRef	Yes	No	An object reference for the DHCP address pool.

**Return value**

Name	Type	Description
dhcpAddressPool	DHCPAddressPool	The DHCP address pool object.

---

**GetDHCPAddressPoolPermitList**

---

Gets a DHCP address pool permit for a given address pool.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpAddressPoolRef	ObjRef	Yes	No	An object reference for the DHCP address pool from which to get the address pool permit list.

**Return value**

Name	Type	Description
dhcpAddressPoolPermits	ArrayOfDHCPAddressPoolPermit	List of DHCP address pool permits. See description of arrays in the <b>Object Types and Properties</b> section.

---

**GetDHCPAddressPools**

---

Gets a list of DHCP address pools from a scope.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpScopeRef	ObjRef	No	No	A reference to the scope containing the address pool(s).
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
dhcpAddressPools	ArrayOfDHCPAddressPool	List of DHCP address pools. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

**GetDHCPBackupStatus**

---

Gets the backup status of a MS DHCP server.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpServerRef	ObjRef	Yes	No	An object reference for the DHCP server.

**Return value**

Name	Type	Description
dhcpBackupExists	boolean	Determines whether a backup exists for the DHCP server.
dhcpBackupTimestamp	DateTime	Contains the timestamp for the latest backup.

---

**GetDHCPClassNames**

---

Gets the names of the DHCP classes from a DHCP server.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpServerRef	ObjRef	Yes	No	An object reference for the DHCP server.

**Return value**

Name	Type	Description
dhcpClassNames	ArrayOfString	An array of DHCP class names.

---

**GetDHCPExclusion**

---

Gets a DHCP exclusion object based on the object reference.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpExclusionRef	ObjRef	Yes	No	An object reference for the DHCP exclusion.

**Return value**

Name	Type	Description
dhcpExclusion	DHCPExclusion	The DHCP exclusion object.

---

**GetDHCPExclusions**

---

Gets a list of DHCP exclusions from a scope.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpScopeRef	ObjRef	No	No	A reference to the scope containing the exclusion(s).
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
dhcpExclusions	ArrayOfDHCPExclusion	List of DHCP exclusions. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

**GetDHCPGroup**

---

Gets a DHCP group object based on the object reference.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpGroupRef	ObjRef	Yes	No	An object reference for the DHCP group.

**Return value**

Name	Type	Description
dhcpExclusion	DHCPExclusion	The DHCP exclusion object.

**GetDHCPGroups**

Gets a list of DHCP groups.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ownerRef	ObjRef	Yes	No	A reference to the object containing the DHCP group(s).
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
dhcpGroups	ArrayOfDHCPGroup	List of DHCP groups. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

**GetDHCPLeaseHistory**

Gets a list of DHCP lease history entries.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
field	DHCPLeaseField	Yes	No	The field by which to filter.
filter	string	Yes	No	Filtering criteria for the records returned.
exactMatch	boolean	No	No	Set to <b>true</b> if the field should match the filter exactly.
collapseRenew	Boolean	No	No	Set to <b>true</b> if renew entries should be collapsed into a single lease history entry.
startDate	DateTime	No	Yes	Date of the first entry to retrieve.
endDate	DateTime	No	Yes	Date of the last entry to retrieve.
Limit	unsignedInt	No	No	The number of records to return.

**Return value**

Name	Type	Description
dhcpLeaseHistoryEntries	ArrayOfDHCPLeaseHistoryEntry	Array of lease history entries fulfilling the search criteria.

---

**GetDHCPLeases**

---

Gets a list of DHCP leases from a scope.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpScopeRef	ObjRef	Yes	No	A reference to the scope containing the leases.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
dhcpLeases	ArrayOfDHCPLease	List of DHCP leases. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

**GetDHCPOptionDefinitions**

---

Gets DHCP option definitions from the object referred to by **ref**.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ref	ObjRef	Yes	No	A reference to the object from which to get the DHCP option definitions.
objType	ObjectType	No	No	The type of the object from which to get the DHCP option definitions.

**Return value**

Name	Type	Description
dhcpOptionDefinitions	ArrayOfDHCPOptionDefinition	The option definitions.

---

**GetDHCPOptions**

---

Gets DHCP options from the object referred to by **ref**.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ref	ObjRef	Yes	No	A reference to the object from which to get the DHCP options.
objType	ObjectType	No	No	The type of the object from which to get the DHCP options.

**Return value**

Name	Type	Description
ticket	unsignedInt	Ticket used for concurrency. Sent back as part of call to SetDHCPOptions and, if there have been changes made to DHCP options for the object since the ticket was returned, the SetDHCPOptions call will fail.
dhcpOptions	ArrayOfDHCPOption	The DHCP options.

**GetDHCPReservation**

Gets a DHCP reservation object based on the object reference.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpReservationRef	ObjRef	Yes	No	An object reference for the DHCP reservation.

**Return value**

Name	Type	Description
dhcpReservation	DHCPReservation	The DHCP reservation object.

**GetDHCPReservations**

Gets a list of DHCP reservations from a scope.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ownerRef	ObjRef	Yes	No	A reference to the DHCP server/group/scope containing the reservations.
objType	ObjectType	No	No	The object type of the owner (i.e. one of DHCP server/group/scope)
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
dhcpReservations	ArrayOfDHCPReservation	List of DHCP reservations. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.



---

## GetDHCPscope

---

Gets a DHCP scope object based on the object reference.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpScopeRef	ObjRef	Yes	No	An object reference for the DHCP scope.

### Return value

Name	Type	Description
dhcpScope	DHCPscope	The DHCP scope object.

---

## GetDHCPscopes

---

Gets a list of DHCP scopes.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

### Return value

Name	Type	Description
dhcpScopes	ArrayOfDHCPscope	List of DHCP scopes. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

## GetDHCPserver

---

Gets a DHCP server object based on the object reference.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpServerRef	ObjRef	Yes	No	An object reference for the DHCP server.

### Return value

Name	Type	Description
dhcpServer	DHCPserver	The DHCP server object.

---

**GetDHCPServers**

---

Gets a list of DHCP servers.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
dhcpServers	ArrayOfDHCPserver	List of DHCP servers. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

**GetDHCPSubclasses**

---

Gets the names of the DHCP subclasses classes for a given DHCP class.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpServerRef	ObjRef	Yes	No	An object reference for the DHCP server.
dhcpClassName	string	Yes	No	The name of the DHCP class for which to retrieve the subclasses.

**Return value**

Name	Type	Description
dhcpSubclasses	ArrayOfString	An array of DHCP subclasses.

---

**ReleaseDHCPLease**

---

Releases a DHCP lease.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpScopeRef	ObjRef	Yes	No	A reference to the scope containing the lease.
lease	IPAddress	Yes	No	The IP address of the lease that should be released.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

Name	Type	Description
ref	ObjRef	Reference to the created record. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

## SetDHCPOptions

---

Sets value of DHCP options for the object referred to by **Ref**.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ref	ObjRef	Yes	No	A reference to the object for which to set the DHCP options.
objType	ObjectType	No	No	The type of the object for which to set the DHCP options.
ticket	unsignedInt	Yes	Yes	Ticket used for concurrency. Returned by GetDHCPOptions command and, if there have been changes made to DHCP options for the object since the ticket was returned, the SetDHCPOptions call will fail.
dhcpOptions	ArrayOfDHCPOption	Yes	Yes	The DHCP options to set.
saveComment	string	No	Yes	Save comment text to associate with the action.

---

## SetDHCPSubclasses

---

Sets value of DHCP subclasses for a given DHCP class.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dhcpServerRef	ObjRef	Yes	No	A reference to the DHCP server.
dhcpClassName	string	Yes	No	The name of the DHCP class.
dhcpSubclasses	ArrayOfString	Yes	Yes	The DHCP subclasses to set. Note that the existing subclasses will be replaced by this value.
saveComment	string	No	Yes	Save comment text to associate with the action.

## DNS specific commands

The following commands are used for managing DNS data. Note that various commands in the generic object management section are used to manage DNS data as well. See the section **Generic object management** for more information on these commands.

---

### AddDNSRecord

---

Adds a new DNS resource record to a zone.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsRecord	DNSRecord	Yes	No	The DNS record to add. See description of DNSRecord in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.
autoAssignRangeRef	ObjRef	No	No	A reference to a range that should be used for automatic assignment of IP addresses.
forceOverrideOfNamingConflictCheck	Boolean	No	No	Set to <b>true</b> to force creation of record regardless of naming conflict check (if enabled in system settings).

#### Return value

Name	Type	Description
ref	ObjRef	Reference to the created record. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

### AddDNSRecords

---

Adds multiple DNS resource records to zones.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsRecords	ArrayOfDNSRecord	Yes	No	An array of the DNS records to add. See description of arrays in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.
autoAssignRangeRef	ObjRef	No	No	A reference to a range that should be used for automatic assignment of IP addresses.
forceOverrideOfNamingConflictCheck	Boolean	No	No	Set to <b>true</b> to force creation of the records regardless of naming conflict check (if enabled in system settings).

**Return value**

Name	Type	Description
objRefs	ArrayOfObjRef	Array of references to the records that were created. The array contains one item for each record in the dnsRecords list. If a record was not successfully created, the array item for that record contains the value {#0;#0}
errors	ArrayOfError	Array of error messages. One error message is generated for each record that is not added. See description of arrays in the <b>Object Types and Properties</b> section.

**AddDNSServer**

Adds a DNS server.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsServer	DNSServer	Yes	No	The DNS server to add. See description of DNSServer in the <b>Object Types and Properties</b> section.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

Name	Type	Description
ref	ObjRef	Reference to the new DNS server. See description of ObjRef in the <b>Object Types and Properties</b> section.

**AddDNSZone**

Creates a new DNS zone.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsZone	DNSZone	Yes	No	The DNS zone to add. See description of DNSZone in the <b>Object Types and Properties</b> section.
masters	ArrayOfIPAddress	No	No	An array containing the IP addresses of the master servers if the new zone is a slave zone.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

Name	Type	Description
ref	ObjRef	Reference to the created zone. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

## GetADPartitions

---

Gets a list of Active Directory partitions.

### Parameters

Name	Type	Required	Nullable	Description
Session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsServerRef	ObjRef	No	No	A reference to the DNS server from which to retrieve the Active Directory partitions.

### Return value

Name	Type	Description
adPartitions	ArrayOfString	List of Active Directory partitions.

---

## GetADSite

---

Gets an AD site based on the object reference.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
adSiteRef	ObjRef	Yes	No	The object reference for the AD Site.

### Return value

Name	Type	Description
adSite	ADSite	The AD Site object.

---

## GetADSites

---

Gets a list of Active Directory sites.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

### Return value

Name	Type	Description
adSites	ArrayOfADSite	List of AD Sites. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

## GetDNSRecord

---

Gets a DNS record object based on the object reference.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsRecordRef	ObjRef	Yes	No	An object reference for the DNS record.

### Return value

Name	Type	Description
dnsRecord	DNSRecord	The DNS record object.

---

## GetDNSRecords

---

Gets a list of DNS resource records for a specific zone.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsZoneRef	ObjRef	Yes	No	Reference to the zone containing the record.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
hideNonAccessibleHosts	boolean	No	No	If set to <b>true</b> the resulting set will not contain records for which the user has no access.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

### Return value

Name	Type	Description
dnsRecords	ArrayOfDNSRecord	List of records. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

## GetDNSServer

---

Gets a DNS server object based on the object reference.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsServerRef	ObjRef	Yes	No	An object reference for the DNS server.

### Return value

Name	Type	Description
dnsServer	DNSServer	The DNS server object.

---

**GetDNSServers**

---

Gets a list of DNS servers.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
dnsServers	ArrayOfDNSServer	List of DNS servers. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

**GetDNSView**

---

Gets a DNS view object based on the object reference.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsViewRef	ObjRef	Yes	No	An object reference for the DNS view.

**Return value**

Name	Type	Description
dnsView	DNSView	The DNS view object.

---

**GetDNSViews**

---

Gets a list of DNS server views for a specified server.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsServerRef	ObjRef	Yes	No	A reference to the DNS server
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.



**Return value**

Name	Type	Description
dnsViews	ArrayOfDNSView	List of DNS server views. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

**GetDNSZone**

Gets a DNS zone object based on the object reference.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsZoneRef	ObjRef	Yes	No	An object reference for the DNS zone.

**Return value**

Name	Type	Description
dnsZone	DNSZone	The DNS zone object.

**GetDNSZones**

Gets a list of DNS zones.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
dnsZones	ArrayOfDNSZone	List of zones. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

## GetRelatedDNSRecords

---

Gets all DNS records that are associated with a specific DNS address (A) record. The associated records that are returned can be of the type CNAME, MX, TXT and AAAA. To determine the association, the data section for all of the record types except for the AAAA record type are matched with the name section of the address record. For AAAA records the name section is compared.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
dnsRecordRef	ObjRef	Yes	No	An object reference for the DNS address record for which to find associated records.

### Return value

Name	Type	Description
dnsRecords	ArrayOfDNSRecord	An array of records associated with the address record.

---

## PlaceInADSite

---

Places an array of IP address ranges into an AD site.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
adSiteRef	ObjRef	Yes	No	The object reference for the AD Site.
rangeRefs	ArrayOfObjRef	Yes	No	An array of ranges that should be placed in the AD site.
saveComment	string	No	Yes	Save comment text to associate with the action.

### Return value

This command has no return value.

---

## RemoveFromADSite

---

Removes an array of IP address ranges from an AD site.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
adSiteRef	ObjRef	Yes	No	The object reference for the AD Site.
rangeRefs	ArrayOfObjRef	Yes	No	An array of ranges that should be removed from the AD site.
saveComment	string	No	Yes	Save comment text to associate with the action.

### Return value

This command has no return value.

## IPAM specific commands

The following commands are used for IP Address Management. Note that various commands in the generic object management section are used to manage IP address data as well. See the section **Generic object management** for more information on these commands.

---

### AddRange

---

Adds a new address range.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
range	Range	Yes	No	The address range to add. See description of Range in the <b>Object Types and Properties</b> section.
discovery	Discovery	No	No	The discovery (ping) schedule that should be set for the address range.
saveComment	string	No	Yes	Save comment text to associate with the action.

#### Return value

Name	Type	Description
ref	ObjRef	Reference to the created range. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

### GetAddressBlocks

---

Gets an array of address blocks based on range reference.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
rangeRef	ObjRef	Yes	No	An object reference for the range.

#### Return value

Name	Type	Description
addressBlocks	ArrayOfAddressBlock	Array of address blocks contained within the specified range.

---

### GetAddressesInUse

---

Gets a list of addresses that are in use within an address range.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
lastseen	Discovery	No	No	Used to filter by the time each address last responded to a ping request.
ranges	ArrayOfRangeTitle	No	No	Names of the address ranges in which to look for addresses.

**Return value**

Name	Type	Description
addresses	ArrayOfAddress	An array of addresses that are considered to be in use in the specified address ranges.

**GetAvailableAddressBlocks**

Gets an array of available address blocks based on a given criteria.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
rangeRef	ObjRef	Yes	No	An object reference for the range.
limit	unsignedInt	No	No	The number of address blocks to return.
mask	unsignedInt	No	No	The minimum size of the address blocks, specified as a subnet mask.
size	unsignedInt	No	No	The minimum size of the address blocks, specified as the number of addresses.
ignoreSubnetFlag	Boolean	No	No	Determines whether the subnet flag should be ignored when determining the size of the address blocks.

**Return value**

Name	Type	Description
addressBlocks	ArrayOfAddressBlock	Array of address blocks fulfilling the given criteria.

**GetIPAMRecord**

Gets an IPAM record object based on the object reference.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
addrRef	ObjRef	Yes	No	An object reference for the IPAM record.
requireRangeRef	ObjRef	No	No	Reference to a range in which the relevant IP address must reside, otherwise the GetIPAMRecord call will fail.

**Return value**

Name	Type	Description
ipamRecord	IPAMRecord	The IPAM record object.

---

**GetIPAMRecords**

---

Gets a list of IPAM records.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
rangeRef	ObjRef	Yes	No	A reference to the range containing the addresses.
includeRelatedDNSRecords	Boolean	No	No	Set to <b>true</b> if IPAM record should include DNS records related to any DNS hosts returned.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.

**Return value**

Name	Type	Description
ipamRecords	ArrayOfIPAMRecord	List of IPAM records. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

**GetNextFreeAddress**

---

Finds the next free address in a range, starting from a specified address.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
rangeRef	ObjRef	Yes	No	A reference to the address range from which to return the free address.
startAddress	IPAddress	No	No	The starting address to use when looking for the next free address.
ping	boolean	No	No	If <b>true</b> the system will ping the address found before returning. If the address responds to a ping request, it is considered to be in use.

**Return value**

Name	Type	Description
address	IPAddress	The next free IP address in the address range.

---

**GetRange**

---

Gets an IPAM range object based on the object reference.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
rangeRef	ObjRef	Yes	No	An object reference for the IPAM range.

**Return value**

Name	Type	Description
range	Range	The range object.

**GetRanges**

Gets a list of ranges.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
folderRef	ObjRef	No	No	Reference to a folder from which to get ranges.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
ranges	ArrayOfRange	List of ranges. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

**GetRangeStatistics**

Gets information on used and free IP addresses in an address range.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
rangeRef	ObjRef	Yes	No	A reference to the range.

**Return value**

Name	Type	Description
used	string	Number of used addresses in the address range.
free	string	Number of free addresses in the address range.

---

## GetRangesWithStatistics

---

Gets a list of address ranges along with information on the number of used and free addresses in the ranges.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
calculateUtilization	Boolean	Yes	No	Determines whether to calculate the number of used and free addresses in the ranges.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	String	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

### Return value

Name	Type	Description
rangesWithStatistics	ArrayOfRangeWithStatistics	List of ranges with information about used and free addresses. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

## GetSubranges

---

Gets a list of ranges contained within a range.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
parentRef	ObjRef	Yes	No	A reference to the range that contains the subranges.
depth	unsignedInt	No	No	Specifies how many levels of subranges should be retrieved.

### Return value

Name	Type	Description
ranges	ArrayOfRange	List of subranges. See description of arrays in the <b>Object Types and Properties</b> section.

---

## GetRoutersForRange

---

Gets a list of routers that have been specified for address collection for the range.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
rangeRef	ObjRef	Yes	No	A reference to the range.

**Return value**

Name	Type	Description
addresses	ArrayOfIPAddress	List of router IP addresses. See description of arrays in the <b>Object Types and Properties</b> section.

**JoinRanges**

Joins a list of address ranges.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ranges	ArrayOfObjRef	Yes	No	The DNS record to add. See description of DNSRecord in the <b>Object Types and Properties</b> section.
copyAccessFromRef	ObjRef	No	No	A reference to an existing range containing the access information that should be used for the new range.
subnet	boolean	No	No	Determines if the range is defined as a subnet.
locked	boolean	No	No	Determines if the range should be set as locked.
autoAssign	boolean	No	No	Determines if it should be possible to automatically assign IP addresses from the range.
customProperties	ArrayOfProperty	No	No	The custom properties for the range.
discovery	Discovery	No	No	The discovery (ping) schedule that should be set for the address range.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

Name	Type	Description
errors	ArrayOfError	Array of error messages. One error message is generated for each record that is not added. See description of arrays in the <b>Object Types and Properties</b> section.

**SetIPAMRecord**

Set an IPAM record object.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ipamRecordBefore	IPAMRecord	Yes	No	The IPAM record as it was before calling SetIPAMRecord.
ipamRecordAfter	IPAMRecord	Yes	No	The IPAM record as it should be after calling SetIPAMRecord.
forceOverrideOfNamingConflictCheck	Boolean	No	No	Set to <b>true</b> to force creation of the records in the new IPAM record regardless of the naming conflict check (if enabled in system settings).
saveComment	string	No	Yes	Save comment text to associate with the action.



**Return value**

Name	Type	Description
errors	ArrayOfError	Errors that occurred when trying to set the IPAM record.

**SetRoutersForRange**

Specifies a list of routers that should be used for address collection for the range.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
rangeRef	ObjRef	Yes	No	A reference to the range.
addresses	ArrayOfIPAddress	Yes	Yes	List of router IP addresses. See description of arrays in the <b>Object Types and Properties</b> section.
recursive	boolean	No	No	Specifies whether a router address should be placed in child subnet if the address belongs to the subnet.

**Return value**

This command has no return value.

**SplitRange**

Splits a single address range into multiple ranges.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
rangeRef	ObjRef	Yes	No	A reference to range that should be split.
cidr	unsignedInt	Yes	No	Determines the size of the resulting ranges, e.g. to split a range into /24 ranges the value 24 would be used.
subnet	boolean	No	No	Determines if the ranges should be defined as a subnet.
locked	boolean	No	No	Determines if the ranges should be set as locked.
autoAssign	boolean	No	No	Determines if it should be possible to automatically assign IP addresses from the ranges.
customProperties	ArrayOfProperty	No	No	The custom properties for the ranges.
discovery	Discovery	No	No	The discovery (ping) schedule that should be set for the address ranges.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

Name	Type	Description
ranges	ArrayOfRange	A array containing the new ranges. See description of arrays in the <b>Object Types and Properties</b> section.

## Generic object management

The following commands are used for managing generic objects. They can be used to manage objects of any type.

---

### GetObjectAccess

---

Gets a access information for an object.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ref	ObjRef	Yes	No	A reference to the object.
objType	ObjectType	No	No	The type of the object.
identityRef	ObjRef	No	No	Object reference for a user or group. If specified, the command only retrieves access for the specified user or group.

#### Return value

Name	Type	Description
objectAccess	ObjectAccess	Object access information. See description of ObjectAccess in the <b>Object Types and Properties</b> section.

---

### GetObjects

---

Gets a list of objects. This command gets an array of object references and returns a list of the objects referred to by the object references.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
objRefs	ArrayOfObjRef	Yes	No	Array of object references.
objType	ObjectType	No	No	The type of the object.

#### Return value

Name	Type	Description
objects	ArrayOfObject	List of objects. See description of arrays in the <b>Object Types and Properties</b> section.

---

### GetProperties

---

Gets properties for an object.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ref	ObjRef	Yes	No	Reference to the object for which to get the properties.
objType	ObjectType	No	No	The type of the object.
onlyCustom	boolean	No	No	Set to <b>true</b> if only the custom properties for the object should be returned. Default value is <b>false</b> .

**Return value**

Name	Type	Description
properties	ArrayOfProperty	List of properties for the object. See description of arrays in the <b>Object Types and Properties</b> section.

**GetPropertyDefinitions**

Gets property definitions for an object type.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
objType	ObjectType	No	No	The type of the object.

**Return value**

Name	Type	Description
propertyDefinitions	ArrayOfPropertyDefinition	List of property definitions for the object. See description of arrays in the <b>Object Types and Properties</b> section.

**HasAccess**

Checks whether the currently logged in user has the specified access for the specified object(s).

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
accessName	string	Yes	No	The name of the access entry for which to check.
objRefs	ArrayOfObjRef	Yes	No	An array of references to the objects for which to check the access.
objType	ObjectType	No	No	The type of the object.

**Return value**

Name	Type	Description
hasAccessEntries	ArrayOfHasAccessEntry	List of entries specifying whether the user has access to each object. See description of arrays in the <b>Object Types and Properties</b> section.

**RemoveObject**

Deletes the object referred to by ref.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ref	ObjRef	Yes	No	Reference to the object that should be deleted
objType	ObjectType	No	No	The type of the object.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

This command has no return value.

---

## RemoveObjects

---

Deletes multiple objects.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
objRefs	ArrayOfObjRef	Yes	No	An array containing references to the objects to delete.
objType	ObjectType	No	No	The type of the object.
saveComment	string	No	Yes	Save comment text to associate with the action.

### Return value

Name	Type	Description
errors	ArrayOfObjectError	Array of errors. One error item is generated for each object that is not removed. See description of arrays in the <b>Object Types and Properties</b> section.

---

## SetObjectAccess

---

Sets access for objects.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
identityAccess	ArrayOfIdentityAccess	Yes	No	The access to set for the objects.
objRefs	ArrayOfObjRef	Yes	No	References to the objects for which to set access.
objType	ObjectType	No	No	The object type of the objects.
saveComment	string	No	Yes	Save comment text to associate with the action.

### Return value

This command has no return value.

---

## SetProperties

---

Sets properties for an object.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ref	ObjRef	Yes	No	Reference to the object for which to set the properties.
objType	ObjectType	No	No	The type of the object.
saveComment	string	No	Yes	Save comment text to associate with the action.
deleteUnspecified	boolean	No	No	Set to <b>true</b> if properties that are not explicitly set should be cleared. Default value is <b>false</b> .
properties	ArrayOfProperty	Yes	No	The properties and property values that should be set for the object.

### Return value

This command has no return value.

## Miscellaneous commands

---

### AddFolder

---

Creates a new folder where objects of a certain type can be placed.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
folder	Folder	Yes	No	The folder to create.
saveComment	string	No	Yes	Save comment text to associate with the action.

#### Return value

Name	Type	Description
ref	ObjRef	Reference to the created record. See description of ObjRef in the <b>Object Types and Properties</b> section.

---

### CalcJoinedRanges

---

Checks whether ranges that have been selected for joining do exist on a subnet boundary.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ranges	ArrayOfObjRef	Yes	No	The folder to create.

#### Return value

Name	Type	Description
from	IPAddress	The starting IP address.
to	IPAddress	The ending IP address.
Subnet	boolean	Determines whether the joined ranges will exist on a subnet boundary.

---

### GetDHCPConcurrentLeases

---

Gets a list of leases that are active in more than one location.

#### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.

#### Return value

Name	Type	Description
dhcpConcurrentLeases	ArrayOfDHCPConcurrentLease	An array of leases active in more than one location.

---

**GetFolder**

---

Gets a folder object based on the object reference.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
folderRef	ObjRef	Yes	No	An object reference for the folder.

**Return value**

Name	Type	Description
folder	Folder	The folder object.

---

**GetFolderForObject**

---

Gets the containing folder for an object based on the object reference.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
ref	ObjRef	Yes	No	A reference to the object for which to retrieve the containing folder.
objType	ObjectType	No	No	The object type.

**Return value**

Name	Type	Description
folder	ObjRef	The folder object.

---

**GetFolders**

---

Gets a list of folders.

**Parameters**

Name	Type	Required	Nilable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
filter	string	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more info on filtering.
offset	unsignedInt	No	No	Specifies the offset to use when listing the records. A value of 0 starts with the first record.
limit	unsignedInt	No	No	The number of records to return.
sortBy	string	No	No	The name of the field to sort by.
sortOrder	SortOrder	No	No	The sort order to use. See description of SortOrder in the <b>Object Types and Properties</b> section.

**Return value**

Name	Type	Description
folders	ArrayOfFolder	List of folders. See description of arrays in the <b>Object Types and Properties</b> section.
totalResults	unsignedInt	Total # of records in the result set.

---

**GetOrphanReverseDNSRecords**

---

Gets a list of PTR records that have no corresponding A record in the system.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.

**Return value**

Name	Type	Description
dnsRecords	ArrayOfDNSRecord	An array of PTR records having no corresponding A record.

---

**Ping**

---

Sends an ICMP ping request to the specified address.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
addrRef	ObjRef	Yes	No	Reference to the IP address to ping.

**Return value**

Name	Type	Description
alive	boolean	Determines whether the host responded to the ping request.
pingTime	unsignedInt	The response time in milliseconds.

---

**PlaceInFolder**

---

Places one or more objects in a folder.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
folderRef	ObjRef	Yes	No	An object reference for the folder.
objRefs	ArrayOfObjRef	Yes	No	List of objects to place in the folder.
saveComment	string	No	Yes	Save comment text to associate with the action.

**Return value**

This command has no return value.

---

**ReportAccess**

---

Returns access information for objects. Used by the web based access report.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
identityRef	ObjRef	Yes	No	Object reference for a user or group. The command only retrieves access information for the specified user or group.
identityType	ObjectType	No	No	The object type of the identity.
objTypes	ArrayOfObjectType	Yes	No	List of object types for which to return access

information.

**Return value**

Name	Type	Description
objectAccess	ArrayOfObjectAccess	Access information for the objects.

**ReportDHCPScopeOptions**

Returns list of scopes that have a specified scope option value. Used by the web based scope options report.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
reportType	DHCPScopeOptionsReportType	Yes	No	Specifies the report type to return.
option	unsignedInt	Yes	No	The option to query for.
vendorClass	string	No	No	The vendor class to use.
userClass	string	No	No	The user class to use.
value	string	No	No	The value to query for.
Filter	String	No	Yes	Filtering criteria for the records returned. See the <b>Filtering</b> section for more information on filtering.

**Return value**

Name	Type	Description
entries	ArrayOfDHCPScopeOptionsReportEntry	List of scopes fulfilling the search criteria.

**ReportHostDiscovery**

Returns list of hosts based on the given criteria. Used by the web based host discovery reports.

**Parameters**

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
type	HostDiscoveryType	Yes	No	Specifies the report type to return.
startDate	DateTime	Yes	No	The start date to use.
subnetFilter	string	No	Yes	The subnet to use if the host list should return results for a given subnet.
limit	unsignedInt	No	Yes	The maximum number of entries to return.

**Return value**

Name	Type	Description
entries	ArrayOfHostDiscoveryReportEntry	List of host entries fulfilling the search criteria.



---

## ReportIPFromClientIdentifier

---

Returns list of reservations and leases across all DHCP servers that match a particular MAC address or client identifier.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
clientIdentifier	string	Yes	No	Specifies the MAC address or client identifier to search for.

### Return value

Name	Type	Description
addresses	ArrayOfIPAddress	An array of IP addresses that are associated with either a DHCP lease or reservation that match the client identifier.
dhcpReservations	ArrayOfDHCPReservation	An array of DHCP reservations that match the client identifier.
dhcpLeases	ArrayOfDHCPLease	An array of DHCP leases that match the client identifier.

---

## SendMail

---

Sends an email message through an SMTP mail server.

### Parameters

Name	Type	Required	Nullable	Description
session	string	Yes	No	Session ID from the <b>Login</b> command.
smtpServer	string	Yes	No	The name of the SMTP mail server.
from	string	Yes	No	The sender's email address.
to	string	Yes	No	The recipient's email address.
cc	string	No	No	The email address of a cc recipient.
bcc	string	No	No	The email address of a bcc recipient.
subject	string	No	No	The subject of the email message.
body	string	Yes	No	The body of the email message.

### Return value

This command has no return value.

## Code Examples

The following code examples contain a demonstration the usage of many of the SOAP commands using PowerShell. For a downloadable version of these scripts and more examples using other languages, see the Men & Mice knowledgebase: <http://kb.menandmice.com> using the keyword "SOAP".

The following examples consist of several PowerShell scripts and then a usage example.

### get-mmAssembly.ps1

This script creates an assembly that can be used to communicate with the service.

```
#####
## Copyright (c) 2010 Men & Mice
##
## get-mmAssembly.ps1
##
## Connect to mm web service, and create a type that allows you to
## interact with it.
##
## Example:
##
##     $mmProxy = proxy server (default "central.autotest.dev.lab")
##     $mmHost = Service host (default "localhost")
#####

param ($mmProxy = "example.server.test", $mmHost = "localhost")
if($myinvocation.mycommand.path -ne $null)
{
    $myPath=(split-path $myinvocation.mycommand.path -parent)
    if($env:path.split(";") -notcontains "$myPath")
    {
        $env:path+= ";$myPath"
    }
}

$url = "http://$mmProxy/_mmwebext/mmwebext.dll?wsdl?server=$mmHost"+"?ns=uni,crm"
[string]$protocol = "Soap"

[void][system.Reflection.Assembly]::LoadWithPartialName("system.web.services")

trap
{
    Write-Host "Received an exception: $_. Exiting."
    exit 1
}

$dcp = new-object system.web.services.discovery.discoveryclientprotocol

$dcp.AllowAutoRedirect = $true
[void]$dcp.DiscoverAny($url)
$dcp.ResolveAll()

# get service name
foreach ($entry in $dcp.Documents.GetEnumerator()) { # needed for Dictionary
    if ($entry.Value -is [System.Web.Services.Description.ServiceDescription]) {
        $script:serviceName = $entry.Value.Services[0].Name
    }
}

$ns = new-Object System.CodeDom.CodeNamespace # "WebServices"

$wref = new-object System.Web.Services.Description.WebReference $dcp.Documents, $ns
$wrefs = new-object system.web.services.description.webreferencecollection
```

## Men & Mice Suite SOAP Command Reference v6.3

```
[void]$wrefs.Add($wref)

$ccUnit = new-object System.CodeDom.CodeCompileUnit
[void]$ccUnit.Namespaces.Add($ns)

$violations = new-object system.web.Services.Description.BasicProfileViolationCollection
$wsill = [system.web.services.WsiProfiles]::BasicProfile1_1

if
([system.web.Services.Description.WebServicesInteroperability]::CheckConformance($wsill,
$wref, $violations)) {

    $webRefOpts = new-object System.Web.Services.Description.WebReferenceOptions
    $webRefOpts.CodeGenerationOptions = "GenerateProperties"

    $csprovider = new-object Microsoft.CSharp.CSharpCodeProvider
    $warnings =
[System.Web.Services.Description.ServiceDescriptionImporter]::GenerateWebReferences(
    $wrefs, $csprovider, $ccunit, $webRefOpts)

    if ($warnings.Count -eq 0) {
        $param = new-object system.CodeDom.Compiler.CompilerParameters
        [void]$param.ReferencedAssemblies.Add("System.Xml.dll")
        [void]$param.ReferencedAssemblies.Add("System.Web.Services.dll")
        $param.GenerateInMemory = $true;
        $param.GenerateExecutable = $false;
        $param.TreatWarningsAsErrors = $false;
        $param.WarningLevel = 4;

        # do it
        $compileResults = $csprovider.CompileAssemblyFromDom($param, $ccUnit);

        if ($compileResults.Errors.Count -gt 0) {
            Write-Warning "Proxy Generation" "Failed."
            foreach ($output in $compileResults.Output) { write-host $output }
            foreach ($err in $compileResults.Errors) { write-warning $err }
        } else {
            $assembly = $compileResults.CompiledAssembly

            if ($assembly) {
                $serviceType = $assembly.GetType($serviceName)
                $assembly.GetTypes() | % { Write-Verbose $_.FullName }
            } else {
                Write-Warning "Failed: `"$assembly is null"
                return
            }
        }

        # Dump the assembly to the pipeline
        $assembly
    } else {
        Write-Warning "Proxy Generation" "Failed."
        Write-Warning $warnings
    }
}

#Script get-mmAssembly.ps1 ends#
```

**mmCreateInstance.ps1**

This is a helper script that creates an instance of the client and adds some attributes to encapsulate the session, host, and proxy. This script uses the previous get-mmAssembly.ps1 so that script needs to be in path (or the same directory as this script).

```
#####
## Copyright (c) 2010 Men & Mice
##
## mmCreateInstance.ps1
##
## Connect to mm web service, and create a type that allows you to
## interact with it.
##
## The created asseby is cached so that type problems from having more than
## one identical assebyes in the shell do not surface.
##
## Example:
##
##     $mmProxy = proxy server (default "central.server.local")
##     $mmHost = Service host (default "localhost")
#####

param ($mmProxy = "central.server.local", $mmHost = "localhost")
if($myinvocation.mycommand.path -ne $null)
{
    $myPath=(split-path $myinvocation.mycommand.path -parent)
    if($env:path.split(";") -notcontains "$myPath")
    {
        $env:path+= ";$myPath"
    }
}

$url = "http://$mmProxy/_mmwebext/mmwebext.dll?wsdl?server=$mmHost"+"?ns=uni,crm"
[string]$protocol = "Soap"

[void][system.Reflection.Assembly]::LoadWithPartialName("system.web.services")

trap
{
    Write-Host "Received an exception: $_. Exiting."
    exit 1
}

## Create the asseby cache, if it doesn't already exist
if(-not (Test-Path Variable:\MenAndMice.com.WebServiceCache))
{
    ${GLOBAL:MenAndMice.com.WebServiceCache} = @{}
}
## Check if there was an assembly from a previous session to this web service.
## If so, return that instead.

$oldAssembly = ${GLOBAL:MenAndMice.com.WebServiceCache}[$url]
if($oldAssembly)
{
    $oldAssembly
    return
}

$dcp = new-object system.web.services.discovery.discoveryclientprotocol

$dcp.AllowAutoRedirect = $true
[void]$dcp.DiscoverAny($url)
$dcp.ResolveAll()

# get service name
foreach ($entry in $dcp.Documents.GetEnumerator()) { # needed for Dictionary
    if ($entry.Value -is [System.Web.Services.Description.ServiceDescription]) {
```

## Men & Mice Suite SOAP Command Reference v6.3

```

        $script:serviceName = $entry.Value.Services[0].Name
    }
}

$ns = new-Object System.CodeDom.CodeNamespace # "WebServices"

$wref = new-object System.Web.Services.Description.WebReference $dcp.Documents, $ns
$wrefs = new-object system.web.services.description.webreferencecollection
[void]$wrefs.Add($wref)

$ccUnit = new-object System.CodeDom.CodeCompileUnit
[void]$ccUnit.Namespaces.Add($ns)

$violations = new-object system.web.Services.Description.BasicProfileViolationCollection
$wsill = [system.web.services.WsiProfiles]::BasicProfile1_1

if
([system.web.Services.Description.WebServicesInteroperability]::CheckConformance($wsill,
$wref, $violations)) {

    $webRefOpts = new-object System.Web.Services.Description.WebReferenceOptions
    $webRefOpts.CodeGenerationOptions = "GenerateProperties"

    $csprovider = new-object Microsoft.CSharp.CSharpCodeProvider
    $warnings =
[System.Web.Services.Description.ServiceDescriptionImporter]::GenerateWebReferences(
    $wrefs, $csprovider, $ccunit, $webRefOpts)

    if ($warnings.Count -eq 0) {
        $param = new-object system.CodeDom.Compiler.CompilerParameters
        [void]$param.ReferencedAssemblies.Add("System.Xml.dll")
        [void]$param.ReferencedAssemblies.Add("System.Web.Services.dll")
        $param.GenerateInMemory = $true;
        # $param.TempFiles = (new-object System.CodeDom.Compiler.TempFileCollection
        "c:\temp", $true)
        $param.GenerateExecutable = $false;
        # $param.OutputAssembly = "$($ns.Name)_${$sdname}.dll"
        $param.TreatWarningsAsErrors = $false;
        $param.WarningLevel = 4;

        # do it
        $compileResults = $csprovider.CompileAssemblyFromDom($param, $ccUnit);

        if ($compileResults.Errors.Count -gt 0) {
            Write-Warning "Proxy Generation" "Failed."
            foreach ($output in $compileResults.Output) { write-host $output }
            foreach ($err in $compileResults.Errors) { write-warning $err }
        } else {
            $assembly = $compileResults.CompiledAssembly

            if ($assembly) {
                $serviceType = $assembly.GetType($serviceName)
                $assembly.GetTypes() | % { Write-Verbose $_.FullName }
            } else {
                Write-Warning "Failed: `"$assembly is null"
                return
            }
        }

        ## Create an assembly and store it in the cache
        ${GLOBAL:MenAndMice.com.WebServiceCache}[$url] = $assembly
        # Dump the assembly to the pipeline
        $assembly
    }
} else {
    Write-Warning "Proxy Generation" "Failed."
    Write-Warning $warnings
}
}
#Script get-mmAssembly.ps1 ends#

```

**mmParamInfo.ps1**

Information on parameters for a given function in the SOAP service can be optioned using the following script. This script is used in the invoke script that wraps a call to the service in an easy to use PowerShell wrapper.

```
#####
## Copyright (c) 2010 Men & Mice
##
## mmParamInfo.ps1
##
## Gets the parameter information on a given method name
##
## Example:
##
##     $methodName = Method name
##     optional $paramName = The name of the method to be called
##     optional $paramPos = position of a parameter in the parameter list if you only
##         require info on one of the parameters.
##     optional $mmServiceInstance = instance of the service, if none is provided then
##         a new service is created.
##     Usage:
##         mmParamInfo.ps1 "GetDNSZones"
#####

param($methodName =$(throw "methodName required."), $paramName = $null, $paramPos =
$null, $mmServiceInstance = $null)

if($myinvocation.mycommand.path -ne $null)
{
    $myPath=(split-path $myinvocation.mycommand.path -parent)
    if($env:path.split(";") -notcontains "$myPath")
    {
        $env:path+= ";$myPath"
    }
}

# Create a default instance from the MM assembly, - if it is not passed in
if ($mmServiceInstance -eq $null)
{
    $mmAssembly = get-mmAssembly.ps1
    $mmServiceInstance = $mmAssembly.CreateInstance("Service")
}

# Get the parameter info on the real method.
foreach ($info in $mmServiceInstance.GetType().GetMethods())
{
    if ($info.Name -eq $methodName)
    {
        if ($paramName -ne $null)
        {
            foreach ($paramInfo in $info.GetParameters())
            {
                if ($paramInfo.Name -eq $paramName)
                {
                    $paramInfo
                }
            }
            Write-Host "The method $methodName does not have a parameter named
$paramName!"
        }
        elseif ($paramPos -ne $null)
        {
            if ($paramPos -le 0 -or $paramPos -ge $info.GetParameters().Length)
            {
                Write-Host "The method $methodName has only
$info.GetParameters().Length many parameters!"
            }
            else

```

## Men & Mice Suite SOAP Command Reference v6.3

```
        {
            $info.GetParameters()[$paramPos]
        }
    else
    {
        $info.GetParameters()
    }
    Exit 0;
}

Write-Host "No method named $methodName was found in the assembly compiled from
$mmServiceInstance.Url!"
trap
{
    Write-Host "Received an exception: $_. Exiting."
    Exit 1
}

# mmParamInfo.ps1 script ends #
```

**mmInvoke.ps1**

This is a wrapper script that can be used to do operations on the server.

```
#####
## Copyright (c) 2010 Men & Mice
##
## mmInvoke.ps1
##
## Invoke a method on a given service instance
## Helper to simplify calls to methods on the soap service
##
## Usage:
##
##     $mmServiceInstance = Initiated service instance
##     $methodName = The name of the method to be called
##     optional parameterlist = parameters for the given method.
## Example:
##     mmInvoke.ps1 $mmi GetDNSZones "name:^organization"
#####

param($mmServiceInstance =$(throw "mmServiceInstance required."), $methodName =$(throw
"methodName required.))

if($myinvocation.mycommand.path -ne $null)
{
    $myPath=(split-path $myinvocation.mycommand.path -parent)
    if($env:path.split(";") -notcontains "$myPath")
    {
        $env:path+= ";$myPath\"
    }
}

# Invoke the desired method and dump into the pipeline
# Get the parameter info on the actual method.
$paramsInfo = mmParamInfo.ps1 $methodName -mmServiceInstance $mmServiceInstance

# Add the session token and then the rest of the parameters (from the args array)
[System.Collections.ArrayList]$argv = New-Object System.Collections.ArrayList
[Void]$argv.Add($mmServiceInstance.mmSession)
foreach ($param in $args )
{
    [Void]$argv.Add($param)
}
# Fill in nulls for the optional parameters
$optionalParamCount = $paramsInfo.Count - $argv.Count
while ( $optionalParamCount-- -gt 0)
{
    [Void]$argv.Add($null)
}

$result = $mmServiceInstance.GetType().GetMethod($methodName).Invoke($mmServiceInstance,
$argv.ToArray())

# Dump the results into the pipeline
$result

trap
{
    Write-Host "Received an exception: $_. Exiting."
    exit 1
}

# mmInvoke.ps1 script ends #
```



## Usage Examples

### Finding and deleting DNSRecords of type A in zone

```
> $mmi = .\mmCreateInstance.ps1 -mmProxy central.server.company.local -mmHost localhost
Please enter password for user administrator: *****
    Note: this creates an instance of the client and prompts for password as none was given.

> mmInvoke.ps1 $mmi "GetDNSZones" "type:A name:^testZone1.test1.com.$"
    Note: This gets a list of all DNSZones of type A that have the name "testZone1.test1.com."
ref          : {#4-#49}
name         : testZone1.test1.com.
dynamic      : False
dynamicSpecified : True
adIntegrated  : False
adIntegratedSpecified : True
dnsViewRef   : {#5-#7}
dnsViewRefs  :
authority    : 1.win2003.autotest.dev.lab.
type         : Master
customProperties : {}
> mmInvoke.ps1 $mmi "GetDNSRecords" "{#4-#49}"
    Note: then all the records for the given zone can be queried.

ref          : {#13-#1589}
name         :
type         : SOA
ttl          : 7200
data         : 1.win2003.autotest.dev.lab. hostmaster 2010050406 28800 7200 604800
7200
comment      :
enabled      : True
dnsZoneRef  : {#4-#49}

ref          : {#13-#1590}
name         :
type         : NS
ttl          :
data         : 1.win2003.autotest.dev.lab.
comment      :
enabled      : True
dnsZoneRef  : {#4-#49}

ref          : {#13-#1591}
name         : rec1
type         : A
ttl          :
data         : 10.5.0.1
comment      :
enabled      : True
dnsZoneRef  : {#4-#49}

> .\mmInvoke.ps1 $mmi "GetDNSRecords" "{#4-#49}" "{#13-#1591}"
    Note: to get a specific record.

ref          : {#13-#1591}
name         : rec1
type         : A
ttl          :
data         : 10.5.0.1
comment      :
enabled      : True
dnsZoneRef  : {#4-#49}

    Note: Create a new record and add it to the zone {#4-#49}.
> $rec = New-Object DNSRecord
> $rec.comment = "test comment"
> $rec.data = "10.5.0.34"
> $rec.dnsZoneRef = "{#4-#49}"
```

## Men & Mice Suite SOAP Command Reference v6.3

```
> $rec.enabled = $true
> $rec.name = "TestName"
> $rec.ref = "."
> $rec.ttl = 900
> $rec.type = "TXT"
> mmInvoke.ps1 $mmi "AddDNSRecord" $rec
{#13-#1680}

> mmInvoke.ps1 $mmi "RemoveObject" "{#13-#1680}"
    Note: Delete the new record (reference {#13-#1680}).
>
```

## Appendix – New and Changed Commands and Definitions

Below, you will find a list of new and changed commands and definitions between versions 6.2.2 and 6.3. A description of the new items can be found in the relevant sections in this document.

### New Data Types

The following data types are new in version 6.3.

- DNSServerType

### New Commands

The following commands are new in version 6.3.

- GetADSite
- GetDHCPBackupStatus
- GetFolderForObject
- GetRangeByIPAddress
- GetRoutersForRange
- Ping
- PlaceInADSite
- RemoveFromADSite
- ReportIPFromClientIdentifier
- SendMail
- SetRoutersForRange

### Changes to Data Types

The following data types have changed in version 6.3.

DNSRecordType

- The **DHCID** record type has been added

## Changes to Object Definitions

The object definitions have changed in version 6.3.

### Range

- The element **childRanges** has been added.

### DHCPLease

- **type** is no longer a required element

### DNSServer

- The elements **proxy**, **type** and **adForest** have been added

### DHCPServer

- **proxy** is no longer nillable

### DHCPScope

- The element **enabled** has been added

### DNSRecord

- The element **aging** has been added

### DHCPReservation

- **name** is now nillable

### IPAMRecord

- The elements **filename**, **serverName** and **nextServer** have been added
- The elements **clientIdentifier** and **deviceType** have been removed

### Device

- The element **deviceType** has been removed

## Changes to Existing Commands

The commands have changed in version 6.3.

### AddDNSRecord

- The element **forceOverrideOfNamingConflictCheck** has been added

### AddDNSRecords

- The element **forceOverrideOfNamingConflictCheck** has been added

### ReportDHCPScopeOptions

- The element **filter** has been added.

#### SetDHCPOptions

- The element **ticket** is now both optional and nillable

#### SetIPAMRecord

- The element **forceOverrideOfNamingConflictCheck** has been added