



- [Home](#)
- [About](#)
- [Index](#)
- [Exchange](#)
- [Lync](#)
- [O365](#)
- [OCS](#)
- [Video](#)
- [Voice](#)

[Posts](#)[Comments](#)

You are here: [Home](#) / [Lync](#) / Viewing Lync User Telephony Modes in PowerShell

Viewing Lync User Telephony Modes in PowerShell

May 22, 2012 by [Jeff Schertz](#) · [2 Comments](#)

A [question](#) was recently asked in the TechNet Lync Server forums which seemed unique enough to warrant a quick blog article based on the fact that it did take a few minutes to explain the response.

When using the basic **Get-CsUser** cmdlet in Lync Server, as with many other “Get” cmdlets, not all of the applicable parameters for the object are listed by default. As some configuration objects may include tens or even hundreds of parameters then listing all by default would be quick way to make these commands useless for day-to-day administrative tasks.

[Tweet](#)

This article shows how to identify exactly which additional parameters are available for a specific cmdlet and then by querying only a few specific parameters how to verify an individual Lync user’s *Telephony* mode configuration based on the results.

Viewing Additional Parameters

There are two ways that this can be accomplished, either by listing all parameter values for the Lync user account (which can return a lot of unwanted information), or by identifying the specific parameters and then querying for only those.

- Using the *Lync Server Management Shell* enter the following cmdlet to list all of the possible parameters for the **Get-CsUser** cmdlet.

```
PS C:\> Get-CsUser Jeff | fl *
```

```
SamAccountName           : jeff
UserPrincipalName         : jeff@schertz.local
FirstName                 : Jeff
LastName                 : Schertz
WindowsEmailAddress       : jeff@mslync.net
<snipped>
```

The [Get-CsUser](#) documentation only lists the common parameters but in the detailed description there is a note which explains that the general [Get-Member](#) PowerShell cmdlet can be used to list all of the possible parameters for a given cmdlet.

- Using the *Lync Server Management Shell* enter the following cmdlet to list all of the possible parameters for the **Get-CsUser** cmdlet.

```
PS C:\> Get-CsUser | Get-Member -MemberType Properties
```

```
TypeName: Microsoft.Rtc.Management.ADConnect.Schema.OCSADUser
```

Name	MemberType	Definition
AcpInfo	Property	Microsoft.Rtc.Management.ADConn...
ArchivingPolicy	Property	Microsoft.Rtc.Management.ADConn...
AudioVideoDisabled	Property	System.Boolean AudioVideoDisabl...
ClientPolicy	Property	Microsoft.Rtc.Management.ADConn...
ClientVersionPolicy	Property	Microsoft.Rtc.Management.ADConn...
ConferencingPolicy	Property	Microsoft.Rtc.Management.ADConn...
DialPlan	Property	Microsoft.Rtc.Management.ADConn...
DisplayName	Property	System.String DisplayName {get;...
DistinguishedName	Property	System.String DistinguishedName...

Enabled	Property	System.Boolean Enabled {get;set;}
EnabledForFederation	Property	System.Boolean EnabledForFedera...
EnabledForInternetAccess	Property	System.Boolean EnabledForIntern...
EnabledForRichPresence	Property	System.Boolean EnabledForRichPr...
EnterpriseVoiceEnabled	Property	System.Boolean EnterpriseVoiceE...
ExternalAccessPolicy	Property	Microsoft.Rtc.Management.ADConn...
FirstName	Property	System.String FirstName {get;set;}
Guid	Property	System.Guid Guid {get;}
HomeServer	Property	Microsoft.Rtc.Management.ADConn...
HostedVoiceMail	Property	System.Nullable`1[[System.Boole...
HostedVoicemailPolicy	Property	Microsoft.Rtc.Management.ADConn...
HostingProvider	Property	System.String HostingProvider {...
Identity	Property	Microsoft.Rtc.Management.ADConn...
IPPBXSoftPhoneRoutingEnabled	Property	System.Boolean IPPBXSoftPhoneRo...
IsValid	Property	System.Boolean IsValid {get;}
LastName	Property	System.String LastName {get;set;}
LineServerURI	Property	System.String LineServerURI {ge...
LineURI	Property	System.String LineURI {get;set;}
LocationPolicy	Property	Microsoft.Rtc.Management.ADConn...
MobilityPolicy	Property	Microsoft.Rtc.Management.ADConn...
Name	Property	System.String Name {get;set;}
ObjectCategory	Property	Microsoft.Rtc.Management.ADConn...
ObjectClass	Property	Microsoft.Rtc.Management.ADConn...
ObjectState	Property	Microsoft.Rtc.Management.ADConn...
OriginatingServer	Property	System.String OriginatingServer...
OriginatorSid	Property	System.Security.Principal.Secur...
PinPolicy	Property	Microsoft.Rtc.Management.ADConn...
PresencePolicy	Property	Microsoft.Rtc.Management.ADConn...
PrivateLine	Property	System.String PrivateLine {get;...
ProxyAddresses	Property	Microsoft.Rtc.Management.ADConn...
PublicNetworkEnabled	Property	System.Boolean PublicNetworkEna...
RegistrarPool	Property	Microsoft.Rtc.Management.ADConn...
RemoteCallControlTelephonyEnabled	Property	System.Boolean RemoteCallContro...
SamAccountName	Property	System.String SamAccountName {g...
Sid	Property	System.Security.Principal.Secur...
SipAddress	Property	System.String SipAddress {get;s...
TargetRegistrarPool	Property	Microsoft.Rtc.Management.ADConn...
TargetServerIfMoving	Property	Microsoft.Rtc.Management.ADConn...
TenantId	Property	System.Guid TenantId {get;set;}
UserPrincipalName	Property	System.String UserPrincipalName...
VoicePolicy	Property	Microsoft.Rtc.Management.ADConn...
WhenChanged	Property	System.Nullable`1[[System.DateT...
WhenCreated	Property	System.Nullable`1[[System.DateT...
WindowsEmailAddress	Property	System.String WindowsEmailAddre...

Notice that this list is much longer than the ~20 parameters that the cmdlet returns by default.

- To query for one of the additional parameters above a Lync user issue the following example cmdlet to return the *ProxyAddresses* value for the Lync user 'Jeff'. The optional *ExpandProperty* switch can be used in front of the desired parameter if the results are longer than what can fit on a single line in the command window.

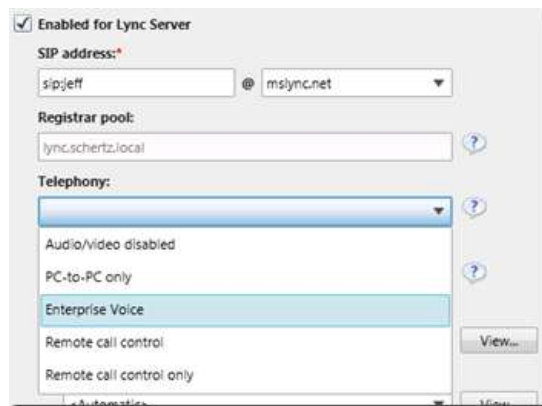
```
PS C:\> Get-CsUser Jeff | Select-Object -ExpandProperty ProxyAddresses

eum:7501;phone-context=ExchangeUM.schertz.local
EUM:jeff@mslync.net;phone-context=ExchangeUM.schertz.local
sip:jeff@mslync.net
SMTP:jeff@mslync.net
```

Verifying Telephony Settings

The Lync user's *Telephony* configuration is actually not a single parameter which would be stamped with a specific value as it appears in the Lync Server Control Panel.

The following screenshot depicts who the *Telephony* setting seems to look like a single setting.



But the actual configuration is stored on the user account by either enabling or disabling a number of different parameters. For the five different telephony settings there are actually up to three different user parameters which may be modified when this setting is changed in the control panel.

These parameters are highlighted in yellow in the Get-Member output shown a few steps previous

Name	MemberType	Definition
AudioVideoDisabled	Property	System.Boolean AudioVideoDisabl...
EnterpriseVoiceEnabled	Property	System.Boolean EnterpriseVoiceE...
RemoteCallControlTelephonyEnabled	Property	System.Boolean RemoteCallContro...

Thus, to identify the configuration of a Lync user simple query all three of these values to identify their values which can be used to deduce the actual *Telephony* mode setting.

- Use the following cmdlet to return only the three desired parameters, using the *Format-List* ('fl') switch to provide for easier reading of the results.

```
PS C:\> Get-CsUser Jeff | Select-Object EnterpriseVoiceEnabled, RemoteCallControlTelephonyEnabled, AudioVideoDisabled | fl
```

```
EnterpriseVoiceEnabled           : True
RemoteCallControlTelephonyEnabled : False
AudioVideoDisabled               : False
```

The results clearly show that his account is enabled for Enterprise Voice as only one of the parameters is set to true. But other settings are not quite as easily to clarify, so the following table can be referenced to identify which Telephony mode the user is configured for based on which parameters are enabled or disabled.

Telephony Mode	EnterpriseVoiceEnabled	RemoteCallControlTelephonyEnabled	AudioVideoDisabled
Audio Video Disabled	False	False	True
PC-to-PC Only (Default)	False	False	False
Enterprise Voice	True	False	False
Remote Call Control	False	True	False
Remote Call Control Only	False	True	True

Note: Enterprise Voice and Remote Call Control modes are mutually exclusive and cannot both be enabled on the same account at the same time.

Filed under [Lync](#) · Tagged with [PowerShell](#)



About Jeff Schertz

Site Administrator

Recent Trackbacks

[Getting Lync users exported to csv based on Conferencing Policy « Myrefspot's Blog](#)

[...] <http://blog.schertz.name/2012/05/viewing-lync-user-telephony-modes-in-powershell/> [...]

[Viewing Lync User Telephony Modes in PowerShell : Jeff Schertz's Blog « JC's Blog-O-Gibberish](#)

[...] Modes in PowerShell : Jeff Schertz's Blog Posted on May 22, 2012 by johnacook <http://blog.schertz.name/2012/05/viewing-lync-user-telephony-modes-in-powershell/> Share this: StumbleUpon Digg Reddit Like this: Like Be the first to like this [...]

Comments (3)

Login

Sort by: [Date](#) [Rating](#) [Last Activity](#)

Kurt Kruger · 650 weeks ago

0

Is it possible to enable a Remote Call Control user using Exchange UM for voicemail to access it using the Visual Voicemail feature in the Lync client? And if yes, how?

Reply



Daniel Clayton · 559 weeks ago

0

Hi Jeff,

Very useful information.

I currently enable new AD users via a script each night:

```
"get-csuser -filter {Enabled -ne $True} -OU "OU=example,DC=example" | Enable-CsUser -RegistrarPool localfqdn -SipAddressType SamAccountName -SipDomain localdomain"
```

Is there a parameter for setting telephony to 'remote call control only'? Currently it defaults to 'PC-to-PC only'

We do not have Enterprise Voice deployed.

Thanks in advance.

Reply [1 reply](#) · active 558 weeks ago[Jeff Schertz](#) 91p · 558 weeks ago

+1

Daniel, the [Set-CsUser](#) cmdlet can be used with the *RemoteCallControlTelephonyEnabled* parameter to enable that.

Reply

Post a new comment

Enter text right here!

Comment as a Guest, or login:

Name

Displayed next to your comments.

Email

Not displayed publicly.

Website (optional)

If you have a website, link to it here.

Subscribe to **Submit Comment**

1347 readers

BY FEEDBURNER



•

•

• Search for:

• Tweets

- A detailed look at the new #Polycom CX5100 for #Lync 2013 <http://t.co/gpSE0Ph8OE>, Dec 24
- How internal #Lync 2013 mobile clients use the Edge server differently <http://t.co/UBjFXuoqDz>, Dec 18
- [@matthewlandis](#) [@UCMadeEasy](#) [@barmach](#) [@Polycom](#) VVX Sidecars are not out yet; coming soon for Lync. As Randy stated - no sidecars for LPE., Nov 15
- A closer look at Boss-Delegate desk phone scenarios with #Lync Shared Line Appearances for VVX Phones <http://t.co/R5Ijj9BWFb>, Oct 31

• Recent Posts

- [January 2014 Lync Users Group Meeting](#)
- [Polycom CX5100 for Lync 2013](#)

- [Lync Mobility Media Paths](#)
- [Shared Line Appearances for VVX Phones](#)
- [Improved RMX Certificate Management](#)

- **Statistics**

- **Total Stats**
 - **168** Posts
 - **2,774** Comments

- **Tag Cloud**

[2013](#) [ABS](#) [Aries](#) [Autodiscover](#) [Certificates](#) [CWA](#) [CX Devices](#) [DHCP](#) [DMA](#) [Edge](#) [Events](#) [HDX](#) [Integration](#) [Interop](#) [Live Meeting](#) [Localization](#) [Lync Phone Edition](#) [Mobility](#) [Normalization](#) [NTLM](#) **[Polycom](#)** [PowerShell](#) [Registration](#) [Reverse Proxy](#) [RMX](#) [RTV](#) [SVC](#) [Tanjay](#) [UCS](#) [VVX](#)

- **Archives**

- [—][2014](#) (1)
 - [—][January](#) (1)
 - [January 2014 Lync Users Group Meeting](#)
- [+] [2013](#) (24)
- [+] [2012](#) (29)
- [+] [2011](#) (26)
- [+] [2010](#) (40)
- [+] [2009](#) (27)
- [+] [2008](#) (15)
- [+] [2007](#) (6)

- **Articles**

- [BPOS Migration Best Practices](#)
- [OCS Certificate White Paper](#)
- [Understanding the Strong Host Model](#)

- **Blogroll**

- [Adam Jacobs](#)
- [Brennon Kwok](#)
- [Chris Norman](#)
- [Dustin Hannifin](#)
- [Elan Shudnow](#)
- [Justin Morris](#)
- [Kevin Peters](#)
- [Mark King](#)
- [Mike Stacy](#)
- [Randy Wintle](#)
- [Tim Harrington](#)
- [Tom Arbuthnot](#)
- [Tom Laciano](#)
- [Tom Pacyk](#)

- **Flickr Photostream**



[Return to top of page](#)

[Posts](#) [Comments](#) · [Log in](#) · Powered by [WordPress](#)

Copyright © 2014 · All Rights Reserved · [Core Blog Theme](#) by [StudioPress](#)