Module 14: Troubleshooting Live Communications Server 2005 with SP1

Contents

Overview

- **Describing Session Initiation Protocol and the Sign-In Process**
- **Tracing Server and Client Side Operations**
- **Using Resource Kit Tools**
- **Troubleshooting Authentication Issues**
- **Interpreting Performance Counters and Errors**

Introduction

You finish this course by investigating and addressing troubleshooting issues with Microsoft® Live Communications Server 2005 with Service Pack 1 (LCS 2005 with SP1). This information will provide you will tools and approaches to investigate service failures and client connectivity problems on LCS 2005 with SP1.
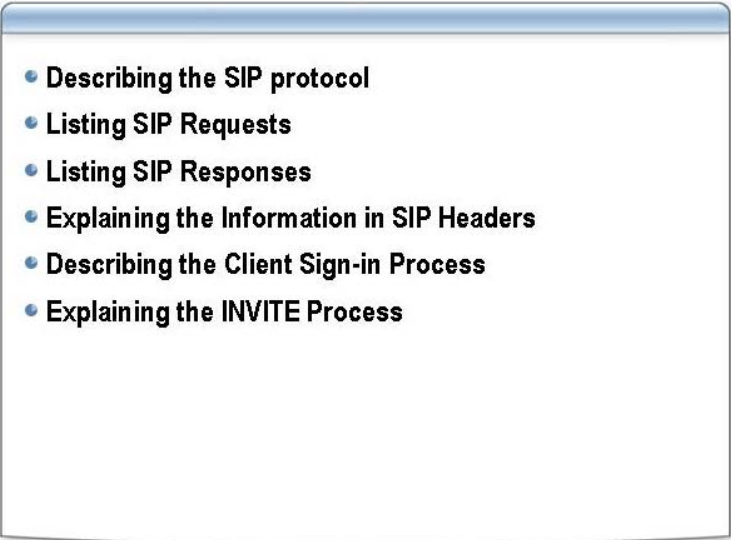
To understand these issues thoroughly, you will start by looking at Session Initiation Protocol (SIP), which is the basis of instant messaging (IM). You then build on this knowledge to review numerous troubleshooting tools and methods, including use of the LCS 2005 with SP1 Resource Kit and the Capacity Planning Toolkit. You finish off this module with a practical example of troubleshooting, where you attempt to use your knowledge to diagnose and fix an inoperative LCS 2005 with SP1 environment.

Objectives

After completing this module, you will be able to:

- Describe Session Initiation Protocol and the sign-in process.
- Trace server and client side operations.
- Use the LCS 2005 with SP1 Resource Kit tools.
- Troubleshoot authentication issues.
- Interpret performance counters and error messages.

Lesson: Describing SIP and the Sign-In Process

- Describing the SIP protocol
- Listing SIP Requests
- Listing SIP Responses
- Explaining the Information in SIP Headers
- Describing the Client Sign-in Process
- Explaining the INVITE Process

Introduction

In Module One, "Introducing Live Communications Server 2005 with Service Pack 1", you learned that SIP is an industry standard, application layer signaling protocol for starting, controlling, and ending communication sessions in an IP-based network. SIP is the core communications protocol used by LCS 2005 with SP1, and is the preferred communications protocol for instant messaging, presence, voice-over-IP telephony, and conferencing across the Internet.

In this lesson, you will look at SIP in greater depth, and identify how SIP enables users to communicate, and the role of SIP in the sign-in process.

Lesson objectives

After completing this lesson, you will be able to:

- Describe the SIP protocol.
- List SIP Requests.
- List SIP Responses.
- Explain the information in SIP Headers.
- Describe the client sign-in process.
- Explain the INVITE process.
- Log SIP communications.

What Is the SIP Protocol?
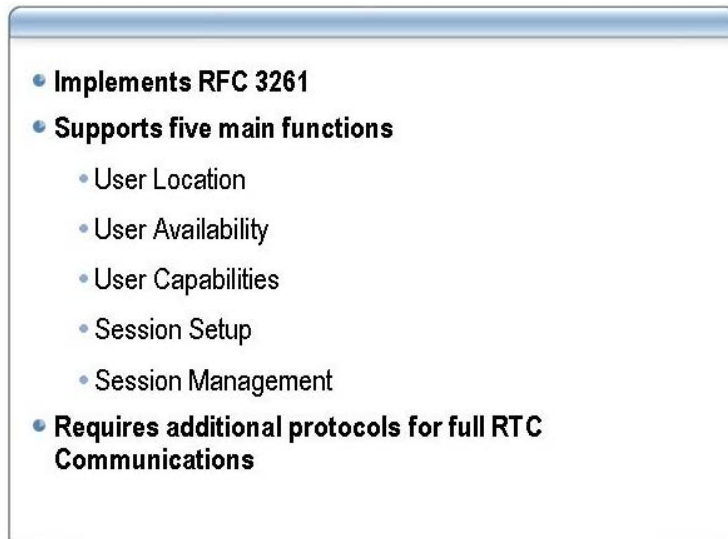
Introduction

SIP provides a central role in LCS 2005 with SP1. Hence, it is important to understand the background, features, and capabilities of SIP to troubleshoot LCS 2005 with SP1 effectively.

IETF Standard

SIP is a request-response protocol that is formally described in the International Engineering Task Force (IETF) reference specification RFC 3261. For more information about RFC 3261, see SIP: Session Initiation Protocol, at http://www.ietf.org/rfc/rfc3261.txt.

SIP supports five main functions in IM conversations:

- **User location**. This function determines the end system that will be used for communications.

- **User availability**. This function registers the willingness of the end user to communicate.

- **User capabilities**. This function determines the media and media parameters to be used for communication.

- **Session setup**. This function establishes session parameters between the calling and the called party.

- **Session management**. This function covers modification of session parameters, invoking services, and terminating the session.

Using SIP, one user can explicitly invite another to join a conversation or multimedia session. A SIP session begins when the second user accepts the invitation. SIP also supports inviting additional users to an already established session.

Because SIP is widely accepted and implemented, it provides a proven technological base for a diverse range of communications. Using a standard protocol for establishing and delivering communications allows for the development of applications that take advantage of existing Internet

technologies while ensuring that the infrastructure can be secured and centrally controlled.

Although SIP sessions can include the sharing of real-time media, SIP itself does not handle the actual media data, such as audio, video, and application sharing. In practical terms, this separation means that SIP and various media protocols can evolve independently.

Additional Protocols

Hence, real-time communications systems may also invoke other protocols to carry out specialist tasks. Examples of these additional protocols include:

- **Real-Time Transport Protocol (RTP).** Transports real-time data and provides Quality of Service (QoS) feedback, as described in RFC 1889.
- **Real-Time Streaming Protocol (RTSP)**. Controls delivery of streaming media, as described in RFC 2326.
- **Media Gateway Control Protocol (MEGACO).** Controls gateways to the Public Switched Telephone Network (PSTN), as described in RFC 3015.
- **Session Description Protocol (SDP)**. Describes multimedia sessions, as set out in RFC 2327.

SIP Client and Servers

A SIP client is an entity that generates requests on behalf of SIP-enabled users. Examples of SIP clients are:

- Microsoft Office Communicator 2005
- Microsoft Windows® Messenger 5.0 and later
- Microsoft Portrait (an application under development from Microsoft Research)

SIP clients are not confined to the desktop computer platform, as SIP applications exist for smartphones, Pocket PCs and may other form factors. Third-party SIP clients can include voice-over-IP (VoIP) softphones, multi-service IM applications, and PC-to-phone clients. SIP applications can implement voice calling, video conferencing, and complex areas such as client relationship management.

RFC 3261 lists three areas of SIP server functionality. These areas are:

- **SIP Registrar Server**. Handles messages that register client locations.
- **SIP Redirect Server**. Provides responses that redirect a request to the correct location.
- **SIP Proxy Server**. Forwards SIP requests and responses between a client and a home SIP server.

LCS 2005 with SP1 provides roles that supply this functionality, receiving the client requests and generating responses. The SIP server may also provide additional services such as authentication and archiving. Numerous SIP servers exist, including both proprietary and open-source, together with public SIP services such as Yahoo!, AOL, and MSN®.

SIP Programming

Microsoft supports SIP programming through the Microsoft Office Live Communications Server 2005 Application Programming Interface (API). This API includes the Request class, which defines a SIP request sent from a client transaction to a server transaction.

Microsoft SIP Processing Language

The Microsoft SIP Processing Language (MSPL) is a scripting language used specifically for filtering and routing SIP messages. Known as "message filters", such scripts (for example, IMFilter.am, the IM URL Filter) are embedded in the application manifests of Live Communications Server applications, such as Office Communicator 2005.

Listing SIP Requests

Name	Function
INVITE	Invites a user
ACK	Acknowledges receipt of a response to an INVITE
CANCEL	Cancels an INVITE
BYE	Requests session termination
REGISTER	Notifies SIP server of new user sign-in
OPTION	Queries server about its capabilities
NOTIFY	Notifies server of event
SUBSCRIBE	Requests notifications for user information
SERVICE	Updates presence and modifies contact list

Introduction

A SIP request includes a request line, several headers, and the message body. The request line contains the Request-URI, which is the address of the next hop and the version of the protocol being used.

Request Commands

SIP requests used in LCS 2005 with SP1 include, but are not limited to, the following:

- **INVITE**. Invites users to a SIP session.

- **ACK**. Acknowledges receipt of a final response to an INVITE request.

- **CANCEL**. Cancels a pending INVITE so long as an ACK request has not already been sent.

- **BYE**. Signals the request to terminate a session.

- **REGISTER**. Informs SIP server that a user has signed in or moved to a new location.

- **OPTION**. Queries a server about its capabilities.

- **NOTIFY**. Sends notification that a particular event has occurred.

- **SUBSCRIBE**. Request notifications for particular user information, such as presence, contact last, and ACL (access control list).

- **SERVICE**. Poll and request updates to presence and modify contact list and ACL.

The SIP Request URI and the To headers are populated with msRTCSIP-LineServer, provisioned in Microsoft Active Directory® directory service.

Protocol Optimizations

Live Communications Server 2005 with SP1 enhances server performance through several SIP protocol optimizations, including:

- **Batch presence subscriptions**. In Live Communications Server 2003, users had to obtain presence information individually for each user in their contact lists. For each user, the client would send a SUBSCRIBE request to the server, and the server would send a NOTIFY in response. In Live Communications Server 2005 with SP1, SUBSCRIBE requests and NOTIFY responses for all the users in a contact list are sent in single batches, reducing network traffic significantly.

- **Piggyback NOTIFY**. The server piggybacks the initial NOTIFY document in the 200 OK of the SUBSCRIBE, thereby avoiding the initial NOTIFY and the following 200 OK response to the NOTIFY.

- **BENOTIFY (Best Effort NOTIFY)**. BENOTIFY is a proprietary SIP method to reduce unnecessary SIP signaling traffic on application servers. Unlike the NOTIFY method, BENOTIFY doesn't require a response. Applications that don't need a response for a NOTIFY request – because they keep track of information synchronization at the application layer, for example – can enhance performance by enabling BENOTIFY. This feature is especially important for deployments with large numbers of clients per server.

- **Whitespace keep-alives**. In order to provide accurate presence information, clients and servers must be able to detect the state of the connection between them on an ongoing basis. In Live Communications Server 2003, connection state detection was accomplished by repeated client re-registration, an expensive operation because it requires the server to write to the database. In Live Communications Server 2005 with SP1, keeping track of the client-server connection is accomplished by the relatively inexpensive means of white-space keep-alives, in which the client makes its availability known by sending CLRFs (carriage return line feeds) to the server. Failure by the server to receive either normal SIP traffic or whitespace keep-alives from the client causes the server to close the connection and inform user services that the client is unavailable.

Listing SIP Reponses

Range	Response Class
100-199	Informational
200-299	Success
300-399	Redirection
400-499	Client Error
500-599	Server Error
600-699	Global Failure

Introduction

The SIP server provides responds to the SIP request with one of a number of possible responses. These responses fall into similar categories to HTTP responses from the Internet Information Service (IIS).

Response Classes

A SIP response is similar in structure to a SIP request, except that it begins with a status line in the place of the request line. The status line includes the protocol version, status code, and text phrase describing the status.

SIP responses fall into the following six classes:

Range	Response Class
100-199	Informational
200-299	Success
300-399	Redirection
400-499	Client Error
500-599	Server Error
600-699	Global Failure

Each Response Class encompasses multiple status codes, for example, 401 "Unauthorized" or 414 "Request-URI too large".

Response Routing

In a LCS 2005 with SP1 SIP network, requests route to end points that represent SIP-enabled Windows user accounts. If there is no user account for a URI in Active Directory, routing is based on static routing tables or federation rules.

Explaining SIP Headers and Protocol Optimizations

Name	Function
To	Provides Address of Recipient
From	Provides Address of Sender
Call-ID	Creates Unique SIP invitation ID
Contact	Lists a user's address (not a SIP Server address)
Record-Route	Ensures routing through a specified server
Route	Lists all Request-URIs in a request path
Via	Lists all LCS servers that routed the request

Introduction

SIP headers are message tags that convey information about a SIP request or response, including, in some instances, about the message body. The Live Communications Server SIP Proxy inspects message headers to determine message processing and routing.

SIP Headers

Among the SIP headers commonly used by Live Communications Server 2005 with SP1 are the following:

■ **To**. Identity of the intended recipient of a request. It takes the form of a SIP URL, which consists of the prefix "Sip:" followed by a user@domain formatted string. The URL in the To header cannot be changed, unlike a Request-URI, which is the address of the next hop and is potentially changed by every server that handles a message.

■ **From**. Identity of the user sending a message.

■ **Call-ID**. Unique identifier of a particular SIP invitation.

■ **Contact**. A user's address, as opposed to the address of the SIP server on which the user is hosted. A server redirecting a message can write the address of the intended recipient in a Contact header returned in a response to the client. Subsequently, the client can contact the recipient directly without having to go through the server.

■ **Record-Route**. A server that proxies a message can add its own URI to the Record Route header to indicate that it wants to remain in the signaling path for all subsequent SIP traffic in the current session. For example, for security reasons, a Live Communications Server 2005 with SP1, Proxy server inserts its Record-Route header into all traffic originating from a corporate branch office to ensure that all responses to such traffic will have to go back through it before crossing the branch office firewall.

■ **Route**. A list of SIP-URIs of all entities in the path of a request. Upon receiving a message, each Live Communications Server removes the Route header that identifies it and forwards the message to the next URI in the list, if there is one.

- **Via**. The address of all Live Communications Servers that have handled a request. Via headers are used to direct responses back to a client by using the same path by which it was sent, but in the opposite direction. A server can also inspect the Via header to determine if it has previously handled a request.

In a Live Communications Server 2005 with SP1 SIP network, requests are routed to end points that represent SIP-enabled Windows user accounts. If there is no user account for a URI in Active Directory, routing is based on static routing tables or federation rules.

Example

The following is an example of a SIP message header.

```
INVITE sip:+14257777777@phones.fabrikam.com;user=phone SIP/2.0
Via: SIP/2.0/TCP 10.2.2.5:16714
Max-Forwards: 70
From:
<sip:alice@fabrikam.com>;tag=0d9280080ada4a1ea504f7d78d434336;
epid=5fc880
096d
To: <sip:+1425557777@phones.fabrikam.com;user=phone>
Call-ID: fdbcb6a6184a4e92a5f001865f84a2c6@10.2.2.5
CSeq: 1 INVITE
Contact:<sip:alice@fabrikam.com:16714
User-Agent: RTC/1.2
Content-Type: application/csta+xml
Content-Disposition: signal; handling=required
Content-Length: n
<?xml version="1.0"?>
<RequestSystemStatus xmlns="http://www.ecmainternational.
org/standards/ecma-323/csta/ed3">
<extensions>
<privateData>
<private>
<lcs:line
xmlns:lcs="http://schemas.microsoft.com/Lcs/2005/04/RCCExtensi
on">tel:+14
257229770;ext=29770</lcs:line>
</private>
</privateData>
</extensions></RequestSystemStatus>
```

What Is the Client Sign-In Process?
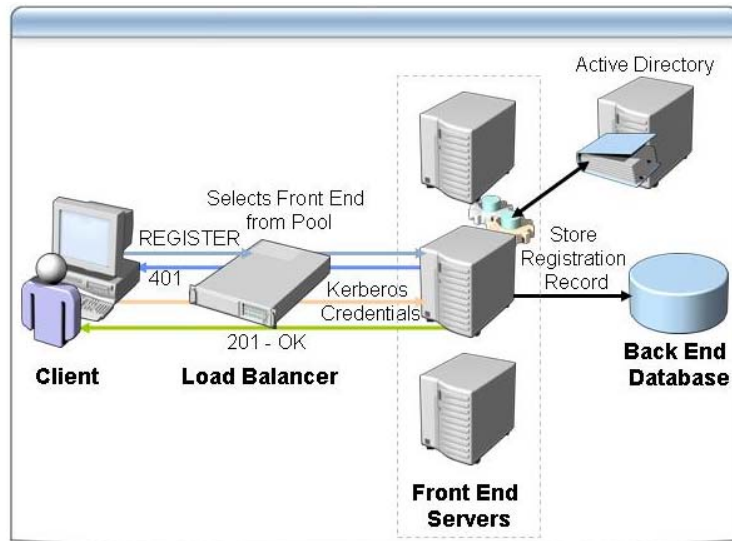
Introduction

When a client signs in, it needs to register with a Standard Edition server or an enterprise pool. This registration process includes authentication with the server.

Note   This example deals with a registration request to an enterprise pool configured with a hardware load balancer. In a Standard Edition server, the registration request is dealt with on the single computer.

Process Steps

The registration process for a client request to an enterprise pool consists of the following steps:

1. The client sends a REGISTER request to the IP address of the enterprise pool.

2. The load balancer selects a front-end server from the enterprise pool, and routes that request to the front-end server.

3. The front-end server issues a 401 Authentication required response.

4. The load balancer routes the 401 response back to the client.

5. The client uses Kerberos authentication to respond with the credentials of the currently logged-on user.

6. The load balancer passes the authentication message back to the front-end server that issued the challenge. The front-end server uses the User Replicator service to authenticate the sign-in credentials, based on information held in Active Directory.

7. The front-end server uses an Open Database Connectivity (ODBC) request to the back end database to store a registration record for the authenticated user. Depending on the user's selected presence settings, the user then appears as online, busy, do not disturb, and so on.

8. The front-end server responds with a 200 OK message.

What Is the INVITE Process?
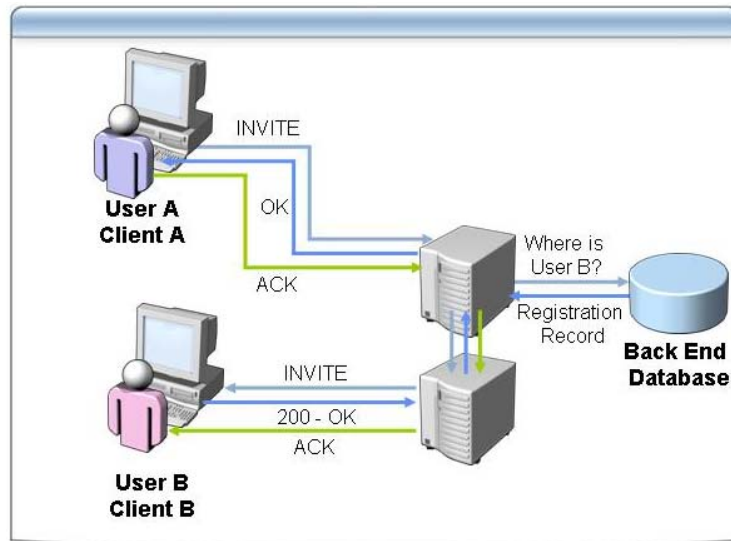
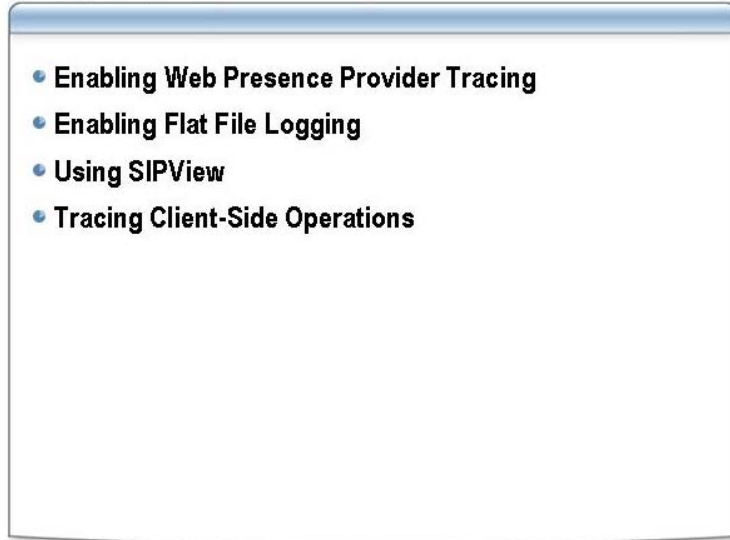| Introduction	After a user has signed in and registered, the user may then want to initiate an IM conversation with someone on his or her contact list, or may search for a user in the Address Book service.

Process Steps

The INVITE process consists of the following steps:

1. The client sends an INVITE request for User B to User A's home server or enterprise pool.

2. The enterprise pool front-end server sends an ODBC query to the back-end SQL database, requesting the registration information for User B.

3. The back-end database responds with the registration record for User B.

4. In this case, User B is currently attached to a different front-end server from User A, so the INVITE request is routed to User B's front-end server.

5. The INVITE request is then sent to User B's client device.

6. User B's client device sends a 200 OK response to User B's front-end server

7. User B's front-end server routes the response through User A's front-end server, and back to User A.

8. User A's client sends out an ACK request.

9. User A's front-end server routes the ACK request to User B's front-end server, and on to User B.

10. The IM conversation can proceed without involving the back-end server.

Server Routing

In more complex environments, the LCS 2005 with SP1 Home Server forwards the INVITE message to the Next Hop fully qualified domain name (FQDN), based on a static routing rule for the URI contained in the INVITE To header. The Next Hop FQDN can be the FQDN of a SIP Proxy or of a SIP/ECMA FE. If the Next Hop is not a Live Communications Server 2005 server, the Next Hop device must be provisioned as a secure link. Once the SIP dialog is initially established, the same dialog path is used in both directions to route subsequent SIP messages.

Lesson: Tracing Server-Side and Client-Side Operations

- **Enabling Web Presence Provider Tracing**
- **Enabling Flat File Logging**
- **Using SIPView**
- **Tracing Client-Side Operations**

Introduction

If LCS 2005 with SP1 operations are not running as expected, tracing server-side and client-side operations is an effective way to check for failures. This process is analogous to checking for message delivery on an e-mail system.

Lesson objectives

After completing this lesson, you will be able to:

- Enable Web Presence Provider Tracing.
- Enable Flat File Logging.
- Use SIPView.
- Use RTCMon to trace client-side operations.
- Configure client debug logging.

Enabling Web Presence Provider Tracing

- Gathers Binary Format Tracing Data
- Uses Encrypted format for Tracing Files
- Requires use of Tracelog or Logman Utilities
- Enforces Security Group Restrictions
  - Administrators
  - Local System
  - Network Service
  - PerfmonLogUsers
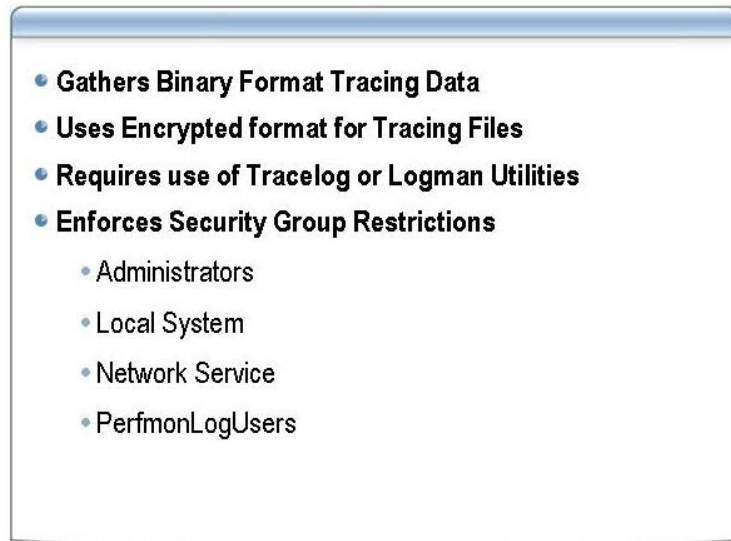
Introduction   Web Presence Provider (WPP) tracing is a binary format that enables network administrators to collect tracing information on their LCS 2005 with SP1 environment. This tracing information is saved into an encrypted file, which Microsoft Product Support Services (PSS) can then interrogate using specialized tools.

Enabling Tracing   Tracing in the Live Communications Server is enabled and controlled by the **tracelog** or **logman** command-line tools.

Security Settings   The default security setting for WPP tracing allows members of the following groups to control the tracing:

- Administrators
- LocalSystem
- NetworkService
- PerfmonLogUsers

Normally, this security setting is good, because the tracing contains only software debugging information. However, in the SIPSTACK component, when the TL_MESSAGE flag is enabled, all traffic on the network gets logged in the tracing files. When encryption is enabled through TLS/SSL, this security setting will defeat the encryption, so Live Communications Server needs to lock down all components that write sensitive data into the log files. Because there is a dependency upon TL_MESSAGE-based logging to provide a support tool, it is not an option to turn off tracing. So, for Live Communications Server, members of the following groups are given the right to control the tracing:

- Local Administrators
- RTC Server Local Groups (only for Standard Edition server or Enterprise Edition server)
- RTC Local Administrators Group

Registry Values

To enable tracing, set the following registry values:

```
HKEY_CURRENT_USER\Software\Microsoft\Tracing\LCAPI\EnableFileT
racing=dword:00000001
HKEY_CURRENT_USER\Software\Microsoft\Tracing\LCAPI\MaxFileSize
=dword:FFFF0000
HKEY_CURRENT_USER\Software\Microsoft\Tracing\LCIMSP\EnableFile
Tracing=dword:00000001
HKEY_CURRENT_USER\Software\Microsoft\Tracing\LCIMSP\MaxFileSiz
e=dword:FFFF0000
```

Trace logs are saved in the folder \Documents and Settings\Tracing.

Enabling Flat File Logging

Introduction

Flat file logging (FFL) provides additional tracing information on what is happening within the LCS environment. FFL logs are in clear text format, and can be read in a text file reader or interrogated by using tools such as FFL Search and SIPView, which are part of the LCS 2005 with SP1 Resource Kit.

Logging Levels

FFL is configured on several levels. These logging levels are:

Logging Level	Detail
1	**Security Logging** – Recommended logging level for all access proxies.
2	**General Logging** – Good for status monitoring without excessive loads.
3	**Detailed Logging** – Provides detail on most events.
4	**Complete Logging** – Provides exhaustive detail on all LCS operations. Best for troubleshooting, but can significantly affect server performance.

Important Logging levels above level 2 are very verbose and can generate very large log files extremely quickly. Only enable higher logging levels if absolutely necessary, and return to a lower logging level after you have completed troubleshooting.

Configuring Logging

To enable and configure FFL, perform the following steps:

1. Click **Start**, point to **Administrative Tools**, and then click **Live Communications Server 2005**.

2. In the console tree, expand **Microsoft Office Live Communications Server 2005**.

3. Expand the **Forest** node.

4. Expand subsequent nodes under the **Domains** node until you reach the domain that the server or pool resides in.

5. Expand the **Live Communications servers and pools** node.

6. Expand the server or pool name.

7. Right-click the FQDN of the server, and then click **Properties**.

8. On the **Logging** tab, click **Enable Logging**.

9. Edit any of the logging information:

   - In the **Logging Level** box, type or select a logging level from **1** through **4**. Each successively higher level logs more information.

   - Click the **Duplicate Application Event** check box if you want Windows NT application events to be logged to the flat file log.

   - In **Log File Folder**, type the directory name where the log file is to be created.

   - In **Create new log when file size reaches**, type the number of megabytes at which time a new log file is created.

   - In **Stop logging when disk usage reaches**, type the percentage of disk capacity at which point logging should stop.

   - In **Continue logging when disk usage drops below**, type the percentage of disk capacity below which logging can resume.

   - Click the **Force Rollover Now** check box to close the existing log file and create a new one. You might want to do this so that you can take a snapshot of the log file for auditing or troubleshooting.

10. Click **Apply**, and then click **OK** to close the **Properties** dialog box.

Using SIPView
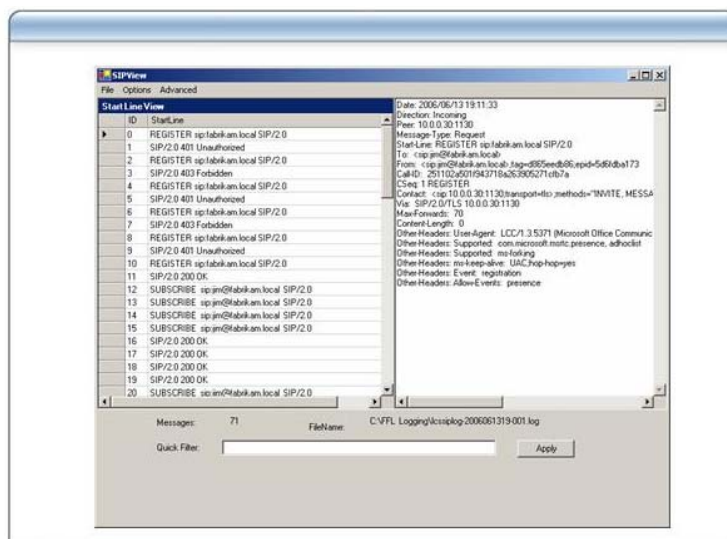
|	
Introduction	SIPView is a specialist tool for tracing SIP conversations from FFL logs. SIPView can open, display, and filter FFL logs through either simple or advanced filtering requests. The tool launches a graphical user interface (GUI).
Obtaining SIPView	SIPView is part of the LCS 2005 with SP1 Resource kit, which you can download from the Microsoft.com Web site, at http://www.microsoft.com/ downloads/details.aspx?FamilyID=d21c38e5-5d8f-44c7-ba17-2cc4f85d8b51&DisplayLang=en. SIPView installs into the \SIPView folder under the LCS 2005 with SP1 Resource Kit installation folder (usually *C:\Program Files\Microsoft LC 2005\ResKit*).
Running SIPView	To run the tool, type [installation folder:]\SIPView\SipView.exe from a command prompt or double-click the tool from within an Explorer window on the Live Communications Server. The tool cannot be run remotely, and the Microsoft .NET Framework needs to be installed on that computer.
	To load a file, click the **File** menu**,** click **Open File**, and browse to a folder that contains a valid Flat File Log file. Alternatively, you can run the tool with the file name specified as a command line parameter.
	SIPView loads the file and renders it in its display. If the file is large, only the first five thousand records will appear. Use the **Advanced** menu to change this setting.
	Use the **Options** menu to filter the data or group the data based on various parameters.

Filtering Logs

1. SIPView allows wildcard filter search on the entire flat file. This search can be accomplished by typing a wildcard text in the Quick Filter text box and hitting the Apply button. The only form of wildcard text that is supported is *Text* (for example, *200*, *REGISTER*).

2. SIPView allows simple SQL queries on the log file. The SQL queries can be applied on the following columns:

   - ID – the identifier field that is shown in the display

   - Method – the SIP method (such as SUBSCRIBE or INVITE)

   - StatusCode – response status code (for example, 200)

   - EdgeIn – the incoming edge

   - EdgeOut – the outgoing edge

   - Message – for full text search on the SIP message. Use wildcard expressions for searching (such as MESSAGE LIKE '*sipuser@sip.example.com*')

   The EdgeIn and EdgeOut parameters can take the following values:

   - 0 – unknown

   - 1 – internal

   - 2 – external

   - 3 – local

   To specify a string parameter, enclose it in single quotes, as in Method='SUBSCRIBE'. For the full set of query capabilities, refer to the DataColumn.Expression function, available from http://msdn2.microsoft.com/en-us/library/system.data.datacolumn.expression.aspx.

Using RTCMon to Trace Client-Side Operations
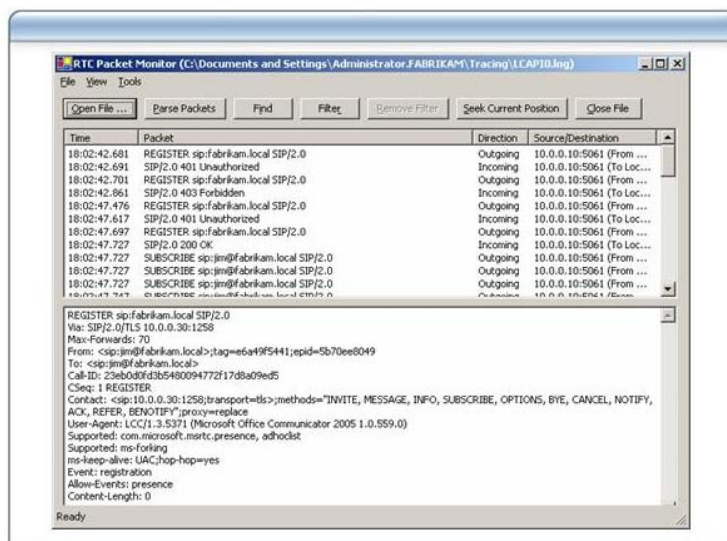
Introduction	The RTCMon tool displays the SIP packets in the log files that are generated by Microsoft Office Communicator (LCAPI*.log) and by versions 1.2 and 1.3 of the RTC (Real-Time Communications) SDK (RTCDLL*.log). The tool launches a GUI.
Obtaining RTCMon	RTCMon is part of the LCS 2005 with SP1 Resource kit, which you can download from the Microsoft.com Web site, at http://www.microsoft.com/ downloads/details.aspx?FamilyID=d21c38e5-5d8f-44c7-ba17-2cc4f85d8b51&DisplayLang=en. RTCMon installs into the \RTCMon folder under the LCS 2005 with SP1 Resource Kit installation folder (usually *C:\Program Files\Microsoft LC 2005\ResKit*).

RTCMon requires the Microsoft .NET Framework v1.1 or later. |
Running RTCMon	On the computer on which the tool is installed, at the command prompt, type [installation folder:]\RTCMon\RTCMon.exe, and then press ENTER. Or in Windows Explorer, go to the installation folder and then double-click RTCMon.exe.
Selecting File Format	By default, RTCMon is set to display the Microsoft Office Communicator log files. On the **Tools** menu, the **LCAPI Format** item has a check mark beside it. To change the view to RTC 1.3 SDK log files, click **LCAPI Format** to clear the check mark. To view Communicator log files again, click **LCAPI Format** to select the check mark.
Viewing Log Files	To view the log files, do one of the following:

- Click Open File, go to the directory where the log files (LCAPI*.log or RTCDLL*.log) are located, and then double-click the file to be displayed.

- Click Parse Packets. RTCMon will determine your client tracing settings as specified in the registry (HKEY_CURRENT_USER\Software\Microsoft\Tracing) for either LCAPI.DLL or RTCDLL.DLL, depending on whether the LCAPI Format item on the Options menu is checked.

Running Commands

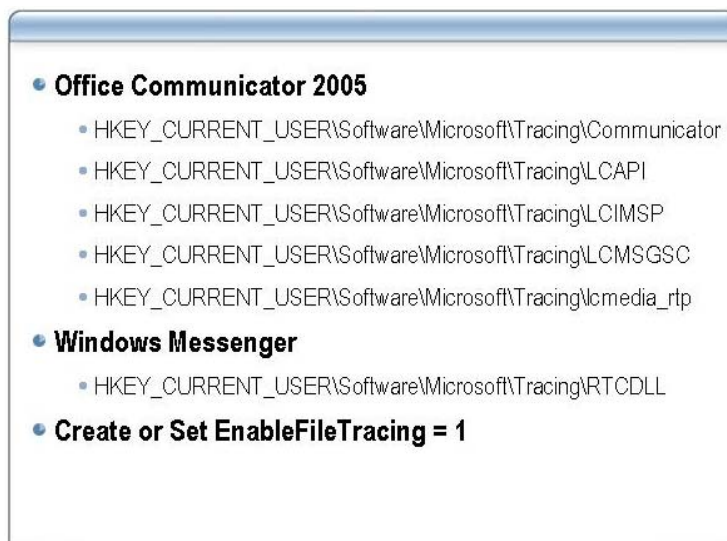RTCMon has several other commands:

- **Parse Packets**. Click this button to incrementally parse the log files. The first time, RTCMon will go through all the generated log files and show the SIP packets contained within them. After that, RTCMon displays only the packets that were sent or received since the last time you performed this operation.

- **Find**. Click to search for a string in the displayed packets.

- **Filter**. Click to view only packets that contain a string that you specify.

- **Remove Filter**. Click to remove the applied filter.

- **Seek Current Position**. Click to move the file marker to the end of the last log file. The behavior is similar to that of the Parse Packets button without the overhead of displaying the packets.

- **Close File**. Click to force RTCMon to release the handle to the log file. RTCMon always holds a handle to the log file, because it needs to determine which packets have already been parsed when the user clicks Parse Packets. By releasing the handle, you force RTCMon to show all SIP packets contained in the generated log files, not just the packets that were sent or received since the last run of Parse Packets.

Note   You can also use RTCMon to automatically parse the log files. On the Tools menu, click Auto Refresh. RTCMon will automatically display any packets that are written to the log files without any user intervention.

Configuring Client Debug Logging

Office Communicator 2005
- HKEY_CURRENT_USER\Software\Microsoft\Tracing\Communicator
- HKEY_CURRENT_USER\Software\Microsoft\Tracing\LCAPI
- HKEY_CURRENT_USER\Software\Microsoft\Tracing\LCIMSP
- HKEY_CURRENT_USER\Software\Microsoft\Tracing\LCMSGSC
- HKEY_CURRENT_USER\Software\Microsoft\Tracing\lcmedia_rtp

Windows Messenger
- HKEY_CURRENT_USER\Software\Microsoft\Tracing\RTCDLL

Create or Set EnableFileTracing = 1

Introduction

To troubleshoot client connectivity from the client side requires you to configure client logging on the client. The client logging settings depend on whether you are using Office Communicator 2005 or Windows Messenger.

Configuration

To enable client debug logging, perform the following steps:

Warning   Serious problems might occur if you modify the registry incorrectly by using Registry Editor or by using another method. These problems might require that you reinstall your operating system. Microsoft cannot guarantee that these problems can be solved. Modify the registry at your own risk.

1. End the Office Communicator 2005 and Windows Messenger processes (if running). To do this, perform these steps:

   a. Right-click the task bar, and then click **Task Manager**.

   b. Click the **Processes** tab, click **communicator.exe** (Office Communicator 2005) or **msmsgs.exe** (Windows Messenger), and then click **End Process**.

   c. When you receive the following message, click **Yes**.

      WARNING: Terminating a process can cause undesired results including loss of data and system instability. The process will not be given the chance to save its state or data before it is terminated. Are you sure you want to terminate the process?

   d. Quit Task Manager.

2. Click **Start**, click **Run**, type **regedit**, and then click **OK**.

3. If you are running Office Communicator 2005, locate and then click the following registry subkey:

   - **HKEY_CURRENT_USER\Software\Microsoft\Tracing\Communicator**

4. If the **EnableFileTracing** registry value does not exist, create it. To do this, perform these steps:

   a. On the **Edit** menu, point to **New**, and then click **DWORD Value**.

   b. In the **New Value #1** box, type **EnableFileTracing**, and then press ENTER.

   c. Right-click **EnableFileTracing**, and then click **Modify**.

   d. In the **Value data** box, type **1**, and then click **OK**.

5. Repeat Step 4 to create the **EnableFileTracing=1** key and value for the following registry entries:

   - **HKEY_CURRENT_USER\Software\Microsoft\Tracing\LCAPI**

   - **HKEY_CURRENT_USER\Software\Microsoft\Tracing\LCIMSP**

   - **HKEY_CURRENT_USER\Software\Microsoft\Tracing\lcmedia_rtp**

6. If you are running Windows Messenger, locate and then click the following registry subkey:

   - **HKEY_CURRENT_USER\Software\Microsoft\Tracing\RTCDLL**

7. If the **EnableFileTracing** registry value does not exist, create it. To do this, perform these steps:

   a. On the **Edit** menu, point to **New**, and then click **DWORD Value**.

   b. In the **New Value #1** box, type **EnableFileTracing**, and then press ENTER.

   c. Right-click **EnableFileTracing**, and then click **Modify**.

   d. In the **Value data** box, type **1**, and then click **OK**.

8. Quit Registry Editor, and then start Office Communicator 2005 or Windows Messenger.

Tracing File Location

For either Office Communicator 2005 or Windows Messenger, the debug tracing file is created in the **%USERPROFILE%\Tracing** folder. To view the contents of this folder, click **Start**, click **Run**, type **%userprofile%\tracing**, and then click **OK**.

Note   After you finish troubleshooting your SIP connection problem, disable debug logging. To do this, set the EnableFileTracing registry values to 0 (zero) for all instances, then stop and restart the client.

Lesson: Using Troubleshooting Tools

- Using LCSWizard
- Using LCSPing
- Using LCSCheck
- Using LCSDiag

Introduction

The LCS 2005 SP1 Resource Kit includes several other troubleshooting tools that are extremely useful for diagnosing issues with the LCS environment. This module concentrates on the four most useful tools, and briefly describes how to use them.

Lesson objectives

After completing this lesson, you will be able to:

- Use LCSWizard.
- Use LCSPing.
- Use LCSCheck.
- Use LCSDiag.

Using LCSWizard
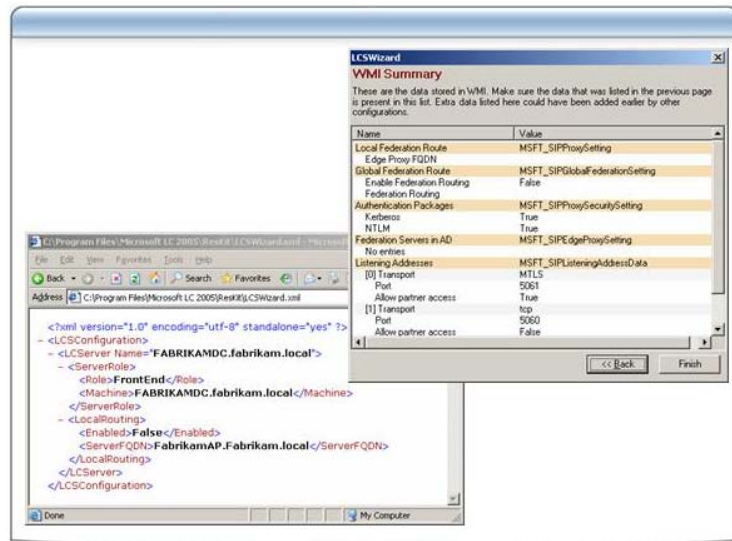
Introduction

The LCSWizard is a tool for configuring settings on LCS 2005 with SP1 Access Proxies, and Standard or Enterprise Edition servers. LCSWizard implements a simple wizard-based GUI.

Obtaining LCSWizard

LCSWizard is part of the LCS 2005 with SP1 Resource kit, which you can download from the Microsoft.com Web site, at http://www.microsoft.com/ downloads/details.aspx?FamilyID=d21c38e5-5d8f-44c7-ba17- 2cc4f85d8b51&DisplayLang=en. LCSWizard installs into the LCS 2005 with SP1 Resource Kit installation folder (usually *C:\Program Files\Microsoft LC 2005\ResKit*).

Note   LCSWizard requires the Microsoft .NET Framework v1.1 or later.

Running LCSWizard

On the computer on which the tool is installed, at the command prompt, navigate to the installation folder for the Resource Kit, type LCSWizard.exe, and then press ENTER. The LCSWizard GUI appears.

The tool launches a wizard-based GUI, and the tool connects to the \root\cimv2 Windows Management Instrumentation (WMI) root to query information about the current LCS setup. The wizard saves settings into an XML configuration file (LCSWizard.xml) and generates results from the query in a text file (LCSWizard.log). You can then use the tool to change settings and save the changes, which are then applied to the LCS server.

Example

The following example shows the contents of the XML configuration file:

```
<?xml version="1.0" encoding="utf-8" standalone="yes" ?>
- <LCSConfiguration>
- <LCServer Name="FABRIKAMDC.fabrikam.local">
- <ServerRole>
  <Role>FrontEnd</Role>
  <Machine>FABRIKAMDC.fabrikam.local</Machine>
  </ServerRole>
- <LocalRouting>
  <Enabled>False</Enabled>
  <ServerFQDN>fabrikamAP.fabrikam.local</ServerFQDN>
  </LocalRouting>
  </LCServer>
  </LCSConfiguration>
```

Using LCSPing
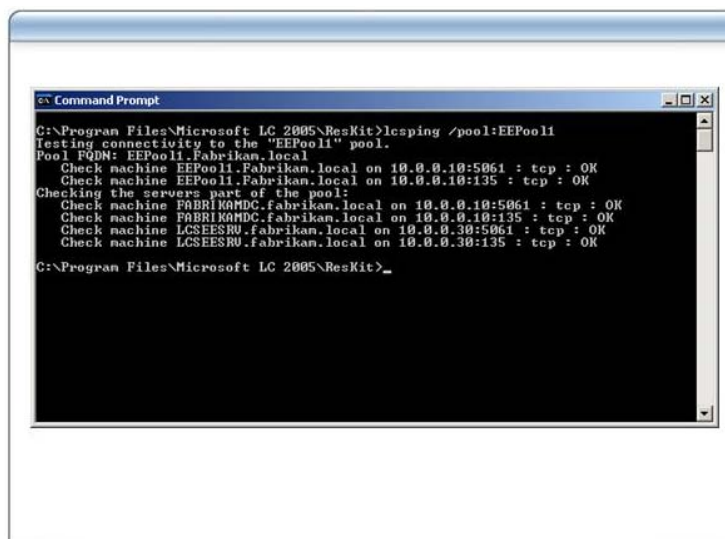
Introduction

LCSPing is a tool for checking client connectivity to a specified Standard Edition server or enterprise pool, and for checking server to server connectivity. LCSPing is a command line utility that you can run from any computer with the LCS administrative tools installed.

Obtaining LCSPing

LCSPing is part of the LCS 2005 with SP1 Resource kit, which you can download from the Microsoft.com Web site, at http://www.microsoft.com/ downloads/details.aspx?FamilyID=d21c38e5-5d8f-44c7-ba17- 2cc4f85d8b51&DisplayLang=en. LCSPing installs into the LCS 2005 with SP1 Resource Kit installation folder (usually *C:\Program Files\Microsoft LC 2005\ResKit*).

Running LCSPing

On the computer on which the tool is installed, start a command prompt, and then navigate to the installation folder for the Resource Kit. Type LCSPing.exe with the required command line switches, and then press ENTER.

LCSPing Syntax

LCSPing takes the following command line switches:

```
lcsping [/s:server [/p:port] [/t:transport]]
  /s:server - specifies the server FQDN that you want to ping
  /p:port - optional, specifies the port to ping. 5061 is the
  default port
  /t:transport - optional, specifies the transport type: tcp
or tls.
  If port is 5061 the default transport is tls
  If port is 5060 the default transport is tcp
lcsping [/wmi [/r:role]]
  /wmi - reads the configuration from WMI
  /r:role - the role of the server. Role can be AP, SE, or EE.
  Default value is AP
lcsping [/pool:poolname]
  /pool:poolname - checks the connectivity to the servers
  part of the given pool
```
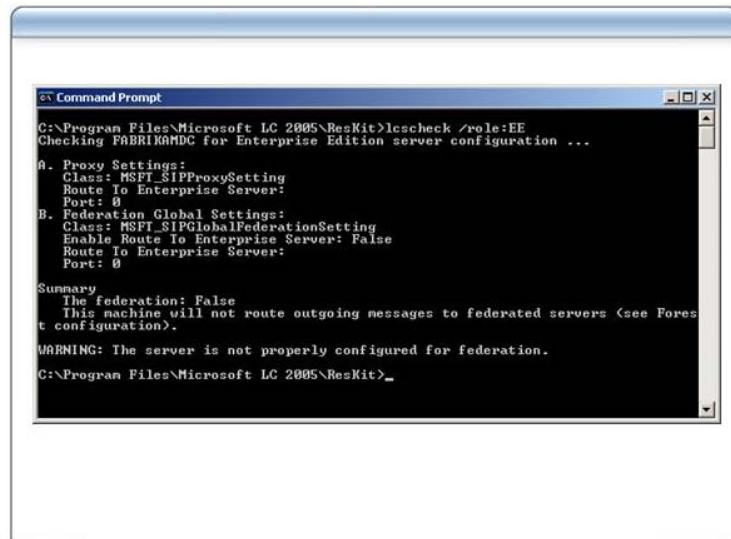
Using LCSCheck

Introduction	LCSCheck is a tool for checking connectivity to an Access Proxy, Standard Edition server or enterprise pool on LCS 2005 with SP1. LCSCheck reads the server configuration from WMI, checks server configuration, and tests both TLS and TCP connectivity to other servers in the LCS environment. LCSCheck is a command line utility.
Obtaining LCSCheck	LCSCheck is part of the LCS 2005 with SP1 Resource kit, which you can download from the Microsoft.com Web site, at http://www.microsoft.com/ downloads/details.aspx?FamilyID=d21c38e5-5d8f-44c7-ba17- 2cc4f85d8b51&DisplayLang=en. LCSCheck installs into the LCS 2005 with SP1 Resource Kit installation folder (usually *C:\Program Files\Microsoft LC 2005\ResKit*).

> **Note**   LCSCheck requires the Microsoft .NET Framework v1.1 or later.

Running LCSCheck	On the computer on which the tool is installed, start a command prompt, and then navigate to the installation folder for the Resource Kit. Type LCSCheck.exe /role: <role>, and then press ENTER.
Role Parameter	The <role> parameter denotes the server role. This parameter can take the following values:

- **AP**. Specifies to test an Access Proxy

- **SE**. Specifies to test a Standard Edition server

- **EE**. Specifies to test an Enterprise Edition server

Using LCSDiag
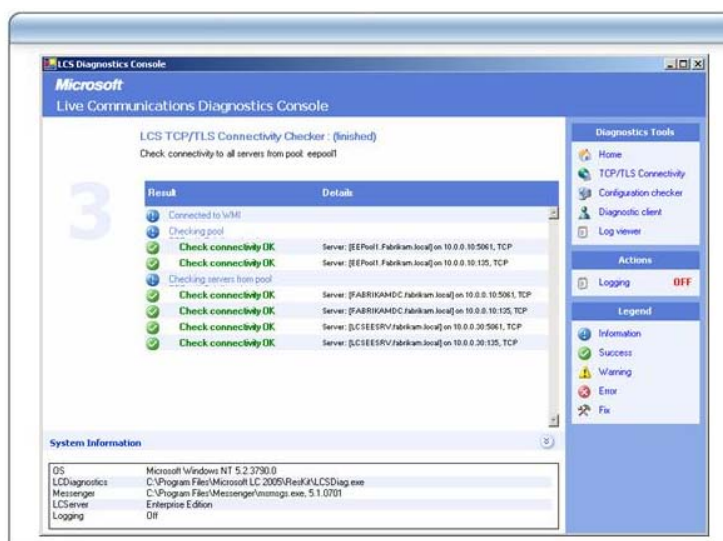
Introduction	The Live Communications Server Diagnostics tool, LCSDiag, enables administrators to check server configuration, connectivity between servers, and client to server connections. LCSDiag is a GUI utility with an integral logging and log-viewing facility.
Obtaining LCSDiag	LCSDiag is part of the LCS 2005 with SP1 Resource kit, which you can download from the Microsoft.com Web site, at http://www.microsoft.com/downloads/details.aspx?FamilyID=d21c38e5-5d8f-44c7-ba17-2cc4f85d8b51&DisplayLang=en. LCSDiag installs into the LCS 2005 with SP1 Resource Kit installation folder (usually *C:\Program Files\Microsoft LC 2005\ResKit*).

Note   LCSDiag requires the Microsoft .NET Framework v1.1 or later.

LCSDiag Credentials	To ensure accurate results, you must run LCSDiag under the same account as the Live Communications Server service. The default name of this account is **DOMAINNAME\LCService**.

Important   The Live Communications Server service account does not have the right to log on locally. You must enable this function by granting LCService the **Allow log on locally** user right, either in the Domain Controller Security Settings group policy, or in the Local Security Settings policy.

After you enable LCService to log on locally, you must either:

- Log on as LCService and start the utility from that logon.
- Run a command prompt under the secondary logon service, and start the utility from that command prompt.
- Create a shortcut to the utility, and set up the shortcut to run with different credentials (recommended if you want to run the tool several times).
- Use the RUNAS command to invoke the utility.

Starting LCSDiag

To start LCSDiag by using the RUNAS command from a command prompt, perform the following steps:

1. On the computer on which the tool is installed, start a command prompt, and then navigate to the installation folder for the Resource Kit.

2. Type **RUNAS /user:DOMAINNAME\LCService LCSDiag.exe**, and then press ENTER.

3. A prompt appears, asking you to type the password for **DOMAINNAME\LCService**. Type in the password.

   Note   The RUNAS command does not display any characters when you type in the password.

4. A message appears **"Attempting to start LCSDiag as user "DOMAINNAME\LCService…"**

5. The LCDiag console appears, showing links to the TCP/TLS Connectivity, Configuration checker, and Diagnostic client.

TCP/TLS Connectivity

The TCP/TLS Connectivity link enables you to test connectivity to:

- **A specific server**. Checks the link to a specified server, typically a Standard Edition server.

- **WMI configured servers**. Checks connectivity to all computers that the current server communicates with.

- **Servers from a pool**. Checks connectivity to an enterprise pool, and to the servers in that pool.

- **A pool FQDN**. Validates that the Pool FQDN is correct.

In each case, you click **Next** to move through a series of steps that carry out the connectivity checks.

Configuration Checker

This test is designed to help you determine whether a specific server is properly configured for settings such as federation and remote access. The test reads WMI values for the server, and validates these values.

Diagnostic Client

The diagnostic client test assists you in debugging the login process for an IM client. To run this test, you must have a valid set of sign-in credentials.

Logging

These three tests can generate .XML formatted logs. To start logging, you must set a log file path.

   Note   If you run LCSDiag under the Live Communications Service service account credentials, you do not have Write access to sub-folders of the %ProgramFiles% or %Windir% folders. Hence, you should save the XML log files in a subfolder of %UserProfile%\My Documents.

Log Viewer

LCDiag enables you to view log files that you have generated from any of the three main tests. The log file viewer also suggests fixes for any detected issues, including links to relevant Microsoft KnowledgeBase articles. If your organization has a support contract with Microsoft PSS, then Microsoft support engineers can use your XML output files to assist with troubleshooting.

Lesson: Troubleshooting Authentication Issues

- Addressing Kerberos Authentication Issues
- Address NTLM Authentication Issues
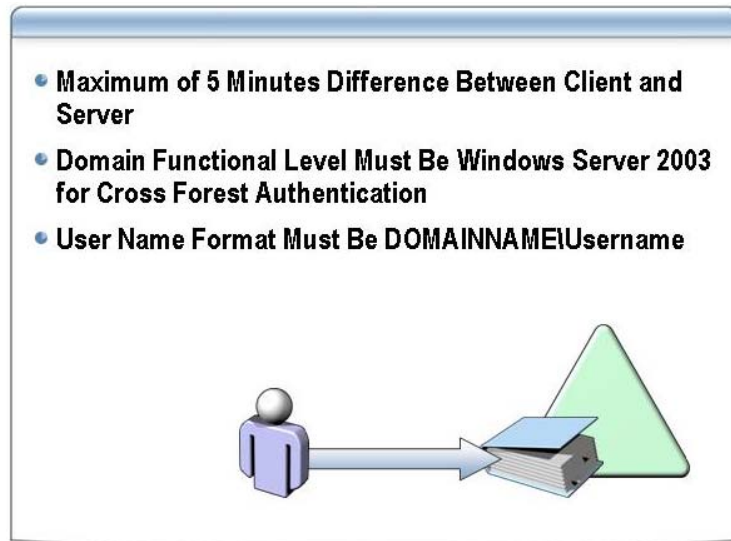- Using the CheckSPN Visual Basic Script

Introduction

Authentication issues are some of the most complex problems to troubleshoot in a LCS Server 2005 with SP1 environment. In this lesson, you review techniques and tools that can address authentication failures, and look the authentication settings in LCS 2005 with SP1.

Lesson objectives

After completing this lesson, you will be able to:

- Address Kerberos authentication issues.
- Address NTLM authentication issues.
- Use the CheckSPN script.

Addressing Kerberos Authentication Issues

- **Maximum of 5 Minutes Difference Between Client and Server**
- **Domain Functional Level Must Be Windows Server 2003 for Cross Forest Authentication**
- **User Name Format Must Be DOMAINNAME\Username**

Introduction

LCS 2005 with SP1 always authenticates users, and that authentication can take place by using either the Kerberos protocol or NTLM. Although Kerberos provides stronger protection of user credentials than NTLM, several issues can arise with this protocol.

Client to server authentication within a Windows Server® 2003 forest uses the Kerberos protocol for authentication between client and server. However, you can configure LCS 2005 with SP1 to use Kerberos protocol, NTLM, or both, depending on your requirements. Outside users that connect through an Access Proxy always require NTLM, as Kerberos does not work through proxies or firewalls.

Time Differences

Kerberos authentication uses timestamps as part of the authentication process. These timestamps require the client and server clocks must not differ by more than five minutes. Any greater time difference prevents users from authenticating by using the Kerberos protocol.

Note   This time difference requirement does not apply to computer clocks that are correctly set but are in different time zones. Hence, a client in London with the time set to 17:00 can synchronize successfully with a server in Redmond that is set to 09:00 on the same day.

Computers that run current versions of Windows include the Windows Time Service. The Windows Time Service is responsible for maintaining synchronization between computers within the same domain.
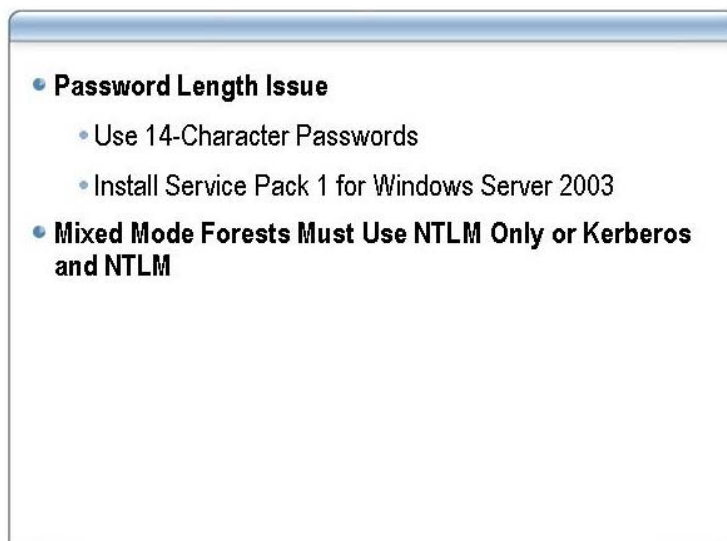
Windows Server 2003 Forests

Kerberos authentication works successfully in an environment where all forests are Windows Server 2003 in native mode and cross-forest trusts are established. However, if the Active Directory environment includes Windows 2000 Server forests or non-native mode Windows Server 2003 forests, then Kerberos authentication does not work between forests. Hence, if the client is in one forest, and the server is in another forest, then Kerberos authentication fails.

User Name Format

When users log on to the IM client, they must type their user names in the format DOMAIN\Username for Kerberos authentication to work. A typical error is for users to type their SIP sign-in name into the user name box.

Addressing NTLM Authentication Issues

- **Password Length Issue**
  - Use 14-Character Passwords
  - Install Service Pack 1 for Windows Server 2003
- **Mixed Mode Forests Must Use NTLM Only or Kerberos and NTLM**

Introduction

NTLM is still supported for remote user access. Request from remote users pass through the Access Proxy to the Director, where the Director authenticates the connection attempt against Active Directory. However, Kerberos authentication requires the user to have access to the Kerberos Key Distribution Server (KDC), which runs on a Windows Server environment as a domain controller. Because direct access from the Internet to intranet domain controllers is highly undesirable, Kerberos authentication is not possible for remote users.

Password Length

Users with a password of less than 14 characters who attempt to authenticate with NTLM may not be able to connect successfully, even though their credentials are correct. The workaround for this issue is to reset their passwords to ones with more than 14 characters, check that they can authenticate, and then change the passwords to a shorter length if necessary.

Note   This issue has been addressed in Service Pack 1 for Windows Server 2003 and later.

Mixed Mode Forests

Kerberos authentication cannot be used where deployments are not exclusively based on Windows Server 2003 forests. If a client in one forest needs to authenticate with a server in another forest, LCS must be set either to **NTLM only** or **Kerberos and NTLM**.

Note   When **Both NTLM and Kerberos** is set on LCS 2005 with SP1, internal clients will always select Kerberos. If Kerberos authentication fails, the client does not attempt to use NTLM.

Checking Service Principal Names
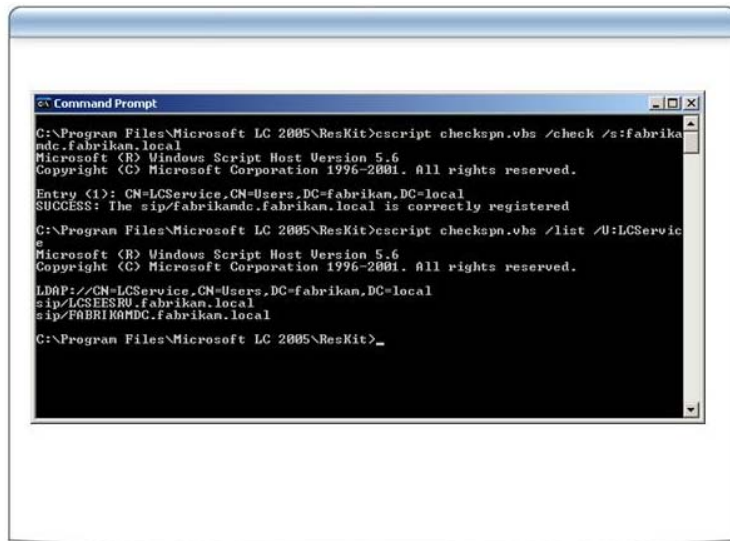
Introduction

The Kerberos protocol authentication used by the Microsoft Office Live Communications Server service for client authentication requires the proper configuration of service principal names (SPNs) within the Active Directory directory service. The SPN is a string identifying the service (Live Communications Server) that a client wants to access. For example, the Live Communications Server will use an SPN of the form SIP/<FQDN of the server>.

For proper operation of Kerberos authentication, the SPN of the Live Communications Server must be registered within Active Directory under the user account where the service runs (typically LCService). If the SPN of the server is registered in multiple accounts, then Kerberos authentication will not operate properly.

Describing CheckSPN

CheckSPN.vbs is a Visual Basic script that can check for and correct issues with SPN naming. This script was designed to address the following two scenarios:

1. To verify that the SPNs have been configured correctly, identify duplicates, and correct any errors in Active Directory.

2. To allow local administrators to install Live Communications Server. The Microsoft Office Live Communications Server 2005 Reference Guide offers steps on how to allow local administrators to install Live Communications Servers. This script is used for that process.

Obtaining CheckSPN

CheckSPN.vbs is part of the LCS 2005 with SP1 Resource kit, which you can download from the Microsoft.com Web site, at http://www.microsoft.com/ downloads/details.aspx?FamilyID=d21c38e5-5d8f-44c7-ba17-2cc4f85d8b51&DisplayLang=en. CheckSPN.vbs installs into the LCS 2005 with SP1 Resource Kit installation folder (usually *C:\Program Files\Microsoft LC 2005\ResKit*).

Running CheckSPN

To run the Checkspn script, open a Command Prompt window and navigate to the folder where you installed the resource kit. At the command prompt, type the following, and then press ENTER:
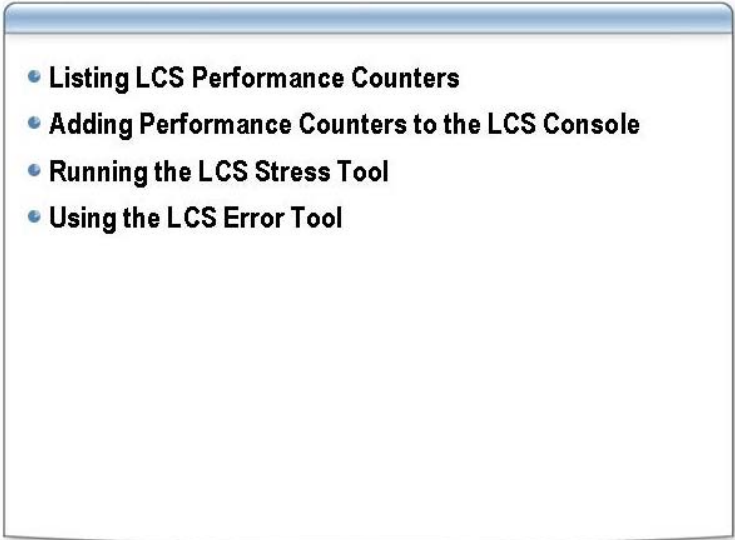
cscript.exe checkspn.vbs <arguments>

Setting Arguments

The Checkspn.vbs script can be run in one of the following four modes:

- **checkspn.vbs /setace**. The /setace switch adds the SchemaGUID property of the User object to the AppliesTo attribute of the Validate SPN extended rights. When this switch is set, the administrator can set an ACE (access control entry) on the Validate SPN right to allow installation of the Live Communications Server by administrators who are not members of the DomainAdmins group. See the Microsoft Office Live Communications Server 2005 Reference Guide for more information about the steps you need to allow non-Domain Administrators to install Live Communications Server.

- **checkspn.vbs /list /u:<admin name>**. The /list switch creates a list of all SPNs registered under a specific user account. The default user account where the Live Communications Server service runs is LCService. List mode is useful for validating that every server has been registered under the proper server account. If the SPN for an installed Standard Edition or Enterprise Edition server is missing from this list, then it should be installed using the Setspn command.

- **checkspn.vbs /check /s:<ServerFQDN>**. The /check switch determines if the SPN for a specific Standard Edition or Enterprise Edition server is registered under one account. The /s switch specifies the server, which is identified by its FQDN. Having the same SPN registered under multiple accounts will cause Kerberos protocol authentication to fail on the client. If there is more than one registration, the Checkspn script prints the user accounts that have this SPN registered. If the server's SPN is registered under multiple accounts, the duplicate SPNs must be deleted until there is exactly one account under which the SPN is registered.

- **checkspn.vbs /del /s:<Server FQDN> /u:<admin name>**. The /del switch deletes the SPN for a specified Standard Edition or Enterprise Edition server from a specified user account. This switch is useful for cleaning up duplicate SPNs. The /s switch specifies the server, which is identified by its FQDN (fully qualified domain name). The /u switch specifies the user account from which the SPN is deleted.

Lesson: Interpreting Performance Counters and Errors

- Listing LCS Performance Counters
- Adding Performance Counters to the LCS Console
- Running the LCS Stress Tool
- Using the LCS Error Tool

Introduction

LCS 2005 with SP1 includes numerous performance counters and event log entries to assist in the process of capacity planning, stress testing and troubleshooting. This final lesson covers how to use performance counters and troubleshooting tools to further diagnose issues with LCS 2005.

Lesson objectives

After completing this lesson, you will be able to:

- List LCS performance counters.
- Add performance counters to the LCS console.
- Run the LCS Stress Tool.
- Use the LCS Error Tool.

Listing LCS Performance Counters
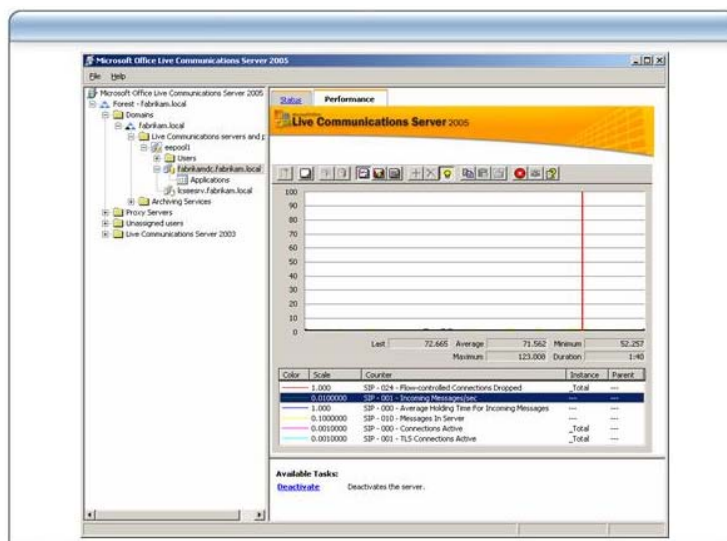
Introduction

LCS 2005 with SP1 includes numerous performance counters that you can check by using the Performance management console. These performance counters can then be used to create graphs in System Monitor, to plot long-term trends in Counter Logs and Trace Logs, or to generate alerts.

LCS 2005 with SP1 adds 32 performance objects, with many of these performance objects having multiple counters. You can display these performance counters either in the Performance console, or in the LCS 2005 with SP1 administrative console.
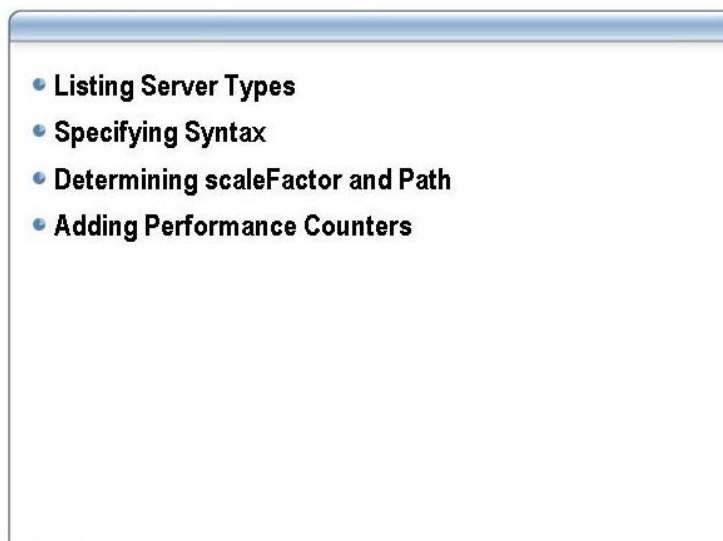
Displaying Performance Counters

To display performance counters in the Performance console, perform the following steps:

1. On the server you want to monitor, click **Start**, select **Run**, and then type **perfmon**.

2. Right-click anywhere in the results pane, and then select **Add Counters**.

3. In the **Add Counters** dialog:

    a. Click **Use local computer counters**.

    b. Select a performance object from the **Performance object** list.

    c. Select one or more counters from the list.

4. Click **Add**.

5. Click **Close**. The newly selected counter appears in the counter list.

6. Click **OK**.

7. Add further counters if required.

Adding Performance Counters to the LCS Console

- **Listing Server Types**
- **Specifying Syntax**
- **Determining scaleFactor and Path**
- **Adding Performance Counters**

Introduction

The **Performance** tab of the LCS 2005 with SP1 management console displays some pre-defined performance counters. You can add, remove, or modify performance counters displayed in the Live Communications Server 2005 management console by editing the file ServerPerfmon.xml, which is located at *%localdrive%*\Program Files\Common Files\Microsoft LC 2005, where *localdrive* is the system drive. To specify additional performance counters, you have to create new entries in the correct format.

Listing Server Types

LCS performance counters enable you to distinguish between server types, or roles. These roles are specified as follows:

```
<counters serverType="EE">
```

Where:

- EE = Enterprise Edition
- SE = Standard Edition
- AP = Access Proxy
- PROXY = Proxy
- ARCH = Archiving

Specifying Syntax

Performance counters are specified as follows:

```
      <counter scaleFactor="0"  path="LC:Arch Service - 01 -
WRITE\Arch Service - 001 - Messages/sec written to DB" />
```

Each performance counter has two attributes, **scaleFactor** and **path**. Both attributes appear in the MMC Performance Monitor.

Scale represents the visual scale of a counter in the Performance Monitor graphic display relative to that of other counters also being displayed. Each counter has a default **scaleFactor** that represents the scale at which the counter display is most likely to be useful, but this value can be adjusted as needed.

The **scaleFactor** value is a power of 10. That is, a **scaleFactor** of "0" equals 10^0 or 1, "1" equals 10^1 or 10, and so on. Negative values are also valid.

Path represents the path to the performance counter.

Determining scaleFactor and Path

You can use the Performance console to determine both the default scale and path of a performance counter. To do this, add the counters into Performance Monitor, as covered in the previous topic, then perform the following additional steps:

1. In the counter list, click the name of the performance counter you have just added.

2. Note the value for that counter under in the **Scale** column. This value is the default scale for that counter. The corresponding **scaleFactor** can be calculated as follows:

   a. If the scale is greater than zero, the **scaleFactor** is the number of zeroes to the left of the decimal point. For example, if the scale is 1000, the **scaleFactor** is "3".

   b. If the scale is less than zero, the **scaleFactor** is the negative of the number of zeroes to the right of the decimal point. For example, if the scale is 0.0001, the **scaleFactor** is –3.

3. Right-click the counter, and then select **Properties**.

4. On the **Data** property page, the path for each counter appears in the **Counters** list. Copy the appropriate path, omitting the initial backward slash (\). In other words, if the path in the **Counters** list is:

```
\Processor( Total)\%Processor Time
```
Then the path you will add to ServerPerfmon.xml will be:

```
Processor( Total)\%Processor Time
```

Adding a performance counter to the LCS management console

To add a performance counter to the LCS 2005 with SP1 console, perform the following steps:

1. Open ServerPerfmon.xml in Notepad or another XML text editor.

2. Beneath the appropriate server type, add a new <counter/> element for the counter added in Performance Monitor, using the format:

```
<counter scaleFactor="#" path="pathname" />
```
Using the example from step 9 in the previous procedure, the new counter element for the Processor Time counter would be as follows:

```
<counter scaleFactor="0" path="
Processor( Total)\%Processor Time" />
```

3. Save ServerPerfmon.xml.

The new performance counter will now appear on the Performance property page of the Live Communications Server 2005 management console.

Running the LCS Stress Tool
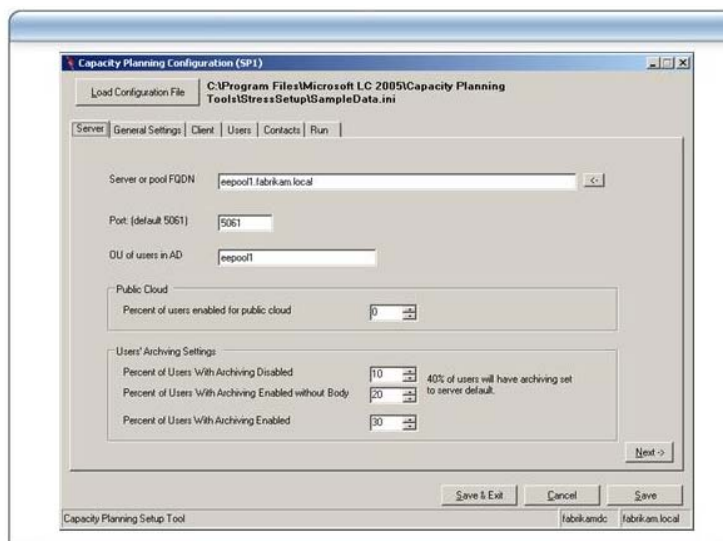
Introduction	The LCS 2005 with SP1 Capacity Planning Toolkit includes the LCS Stress tool, which enables you to generate test loads on your LCS environment. These test loads can then be used to create baseline performance settings, to measure the largest number of users your environment will support, and to plan for future capacity.
Obtaining the Toolkit	The LCS 2005 with SP1 Capacity Planning Toolkit is available for download from the Microsoft.com Web site, at http://www.microsoft.com/downloads/details.aspx?familyid=F249A48A-FC42-4D30-B60B-CB91BF8F2191&displaylang=en. Clicking on this link downloads the LCStress.msi installer file.
Installing the Toolkit	After you have downloaded the LCSStress.msi file, double-click on the file to start the installation process. The default installation location is C:\Program Files\Microsoft LC 2005\Capacity Planning Tools. The setup routine also creates a program group in the Start menu.
Configuring LCSStress	To create a loading on the LCS server, perform the following steps:

1. Click **Start**, point to **All Programs**, point to **Live Communications Server 2005 Capacity Planning Tools**, and click **Stress Setup**.

2. Click the **Load Configuration** File button to open a pre-existing configuration file, or create a new one.

3. On the **Server** tab, type the correct values for the **Server or pool FQDN**, **Port** and **OU of users in AD**. Do not use an existing OU for testing. Set the Public Cloud and Archiving settings as appropriate for your organization, and then click **Next**.

4. On the **General Settings** tab, type values for fields such as the username prefix, the password, the number of users and the SIP and Active Directory domain information.

5. On the **Client** tab, configure how often your users will carry out IM sessions, searches and how many run Windows Messenger.

6. In the **Users** tab, click the **Create Users** button to generate the number of user accounts.

7. In the **Contacts** tab, click **Create Contacts** to generate the contacts for your users.

Note   If you receive an error message at this point, check that you have twice as many users as you have contacts per user, click the **Save and Exit** button, then restart the tool, return to the Contacts tab and retry the operation.

---

8. In the Run tab, note the syntax for the client command, then click **Save and Exit**.

Loading the Server

After you have generated the configuration INI file, you run the client component from a command line. The client component is the LcsUserStress utility, located in the \Installation Folder\UserStress folder.

The syntax for the command is the following, where the INI file is the file you saved from the LCS Stress tool.

LcsUserStress –c INI file

You then see the following output from the command prompt:

```
C:\Program Files\Microsoft LC 2005\Capacity Planning
Tools\UserStress>lcsuserstress -c sampledata.ini
Microsoft (R) LCS User Stress V2.0
Copyright (C) Microsoft Corporation 2001-2005. All rights
reserved.
Adding server eepool1.fabrikam.local:0;transport=2
Adding IP Address 0 0x0a00000a
Total logons: 1000
```

Viewing the Server Load

If you now switch to the LCS console and click the Performance tab, you should see small changes to the performance counters. Click each counter in turn, and view the figures underneath the graph.
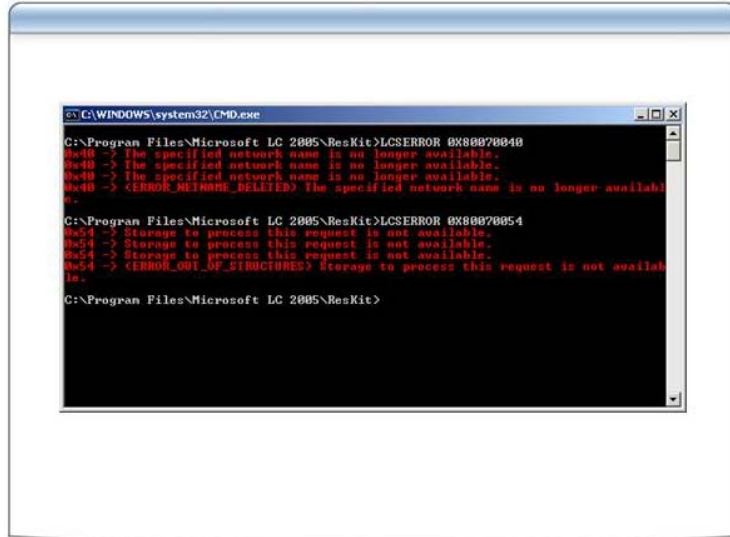
To terminate the test, switch back to the command prompt, and press CTRL+C. The utility then starts the shutdown process.

Note   The shutdown process generates a significant load on the server.

Using the LCSError Tool

Introduction

The Microsoft Office Live Communications Server 2005 command-line script tool, LCSError.exe, helps in troubleshooting event log errors that do not display descriptive text. The tool attempts to map the error code to a description of the error. When possible, it prints the cause of the error and recommends a resolution. For the full functionality of LCSError to be available, run the tool on a computer that has Live Communications Server installed.

Warning   If you install the Resource Kit on a computer that is not running Live Communications Server 2005, you may see warnings with messages like the following when you use LCSError.exe: "Unable to load resource dll LCDeployR.dll." Also, some error codes may return no matches.

Obtaining LCSError

LCSError is part of the LCS 2005 with SP1 Resource kit, which you can download from the Microsoft.com Web site, at http://www.microsoft.com/downloads/details.aspx?FamilyID=d21c38e5-5d8f-44c7-ba17-2cc4f85d8b51&DisplayLang=en. LCSError installs into the LCS 2005 with SP1 Resource Kit installation folder (usually *C:\Program Files\Microsoft LC 2005\ResKit*).

Running LCSError

On the computer on which the tool is installed, start a command prompt, and then navigate to the installation folder for the Resource Kit. Type LCSError.exe "errorcode", and then press ENTER.

For example, if LCS displays the 0x80070054 error, type

```
LCSError 0x80070054
```

The utility returns the following output:

```
0x54 -> Storage to process this request is not available.
0x54 -> (ERROR_OUT_OF_STRUCTURES) Storage to process this
request is not available.
```

Lab 14: Troubleshooting LCS 2005 with SP1 (Optional)

Introduction

Fabrikam and NWTraders have entered into a partnership arrangement that requires selected employees from both companies to co-operate closely. Both companies run LCS 2005 with SP1, so could use Federation to implement secure IM communications. Matt Dawson and Holly Holt have been tasked with the job of setting up Federation between the organizations.

Unfortunately, for Holly and Matt, this process is proving to be more challenging than they expected…

Objectives

After completing this lab, you will be able to:

■ Diagnose and fix common incorrect configurations that affect LCS 2005 with SP1 deployments

■ Troubleshoot communication issues between federated organizations

■ Identify security problems and improper certificate configurations

Estimated time to complete this lab: **60 minutes**

ℹ **Note: You should work in pairs on this lab, if possible.**

Ask your instructor for assistance if you are not making any progress after 40 minutes.

Network Topology

The labs in this course use virtual computers. In order to configure the virtual computers to be usable in a lab environment the network topology has been substantially modified from a typical network configuration. The lab configuration combines many server roles in non-standard ways that are not recommended and are generally not viable in a production network. The network topology used in these labs is shown in the following figures.
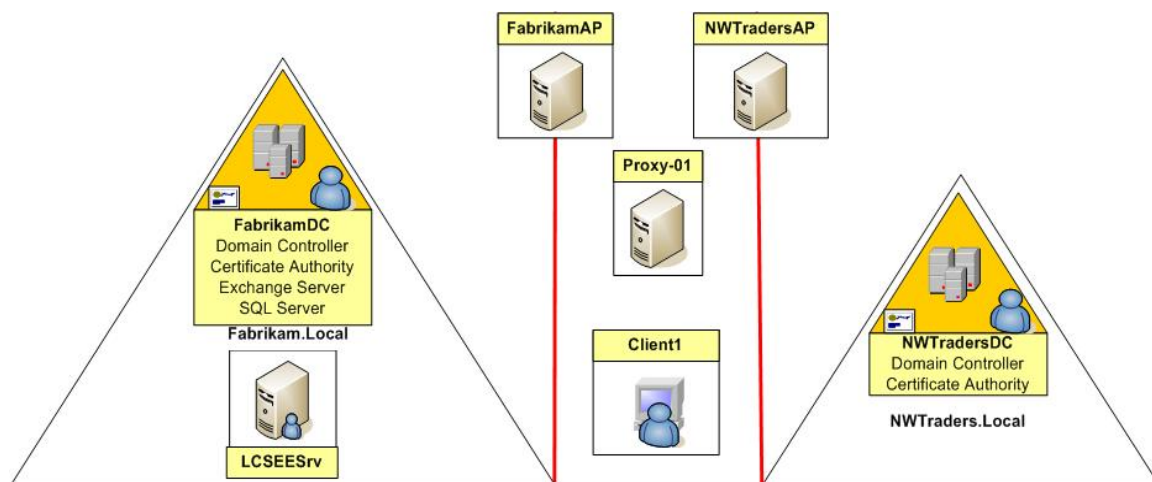
Physical Network Topology
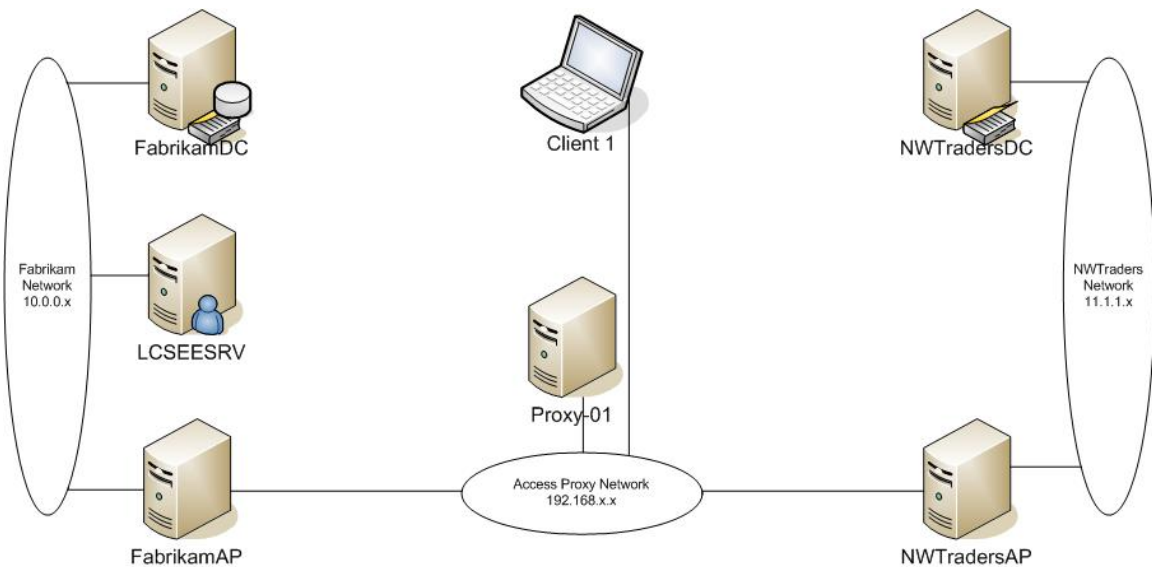
Virtual PC Image to Computer NetBIOS Name Mappings

The following table shows the mapping between the VPC images and the computer NetBIOS names for this lab. Please ensure you use the correct VPC image from the VPC console to start the lab.

VPC Configuration Name	Computer NetBIOS Name
7034A-FabrikamDC-D	FabrikamDC
7034A-FabrikamAP-D	FabrikamAP
7034A-LCSEESRV-D	LCSEESRV
7034A-NWTradersDC-D	NWTradersDC
7034A-NWTradersAP-D	NWTradersAP
7034A-Proxy01-D	Proxy-01
7034A-Client1-D	Client1

Important: You should start all these virtual PC images prior to commencing the labs in this module.

On 7034A-FabrikamDC-D and 7034A-NWTradersDC-D, a Service Control Manager message box may appear, with the following message: At least one service or driver failed during system startup. Use Event Viewer to examine the event log for details. If this message appears, click OK, and continue. The message refers to the Kerberos Key Distribution Center service. However, the service appears to start properly. This error message is not part of the troubleshooting exercise.

At the end of the lab, close down all virtual PC images without saving changes.

Exercise 1
Troubleshooting Federation Issues

Scenario

Matt and Holly have carried out the instructions for setting up Federation between their organizations over the weekend. However, they did not quite get to the testing phase, and on Monday morning, they have come in to the office early to complete the process. They both start up their computers and log on to Office Communicator 2005, hoping to establish communications.

Description

In this lab, you have the opportunity to troubleshoot the Federation connection between Fabrikam and Northwind Traders. You should troubleshoot the connection between Fabrikam and Northwind Traders by looking at the following areas:

- Networking

- DNS

- Certificate

- LCS service

- LCS settings

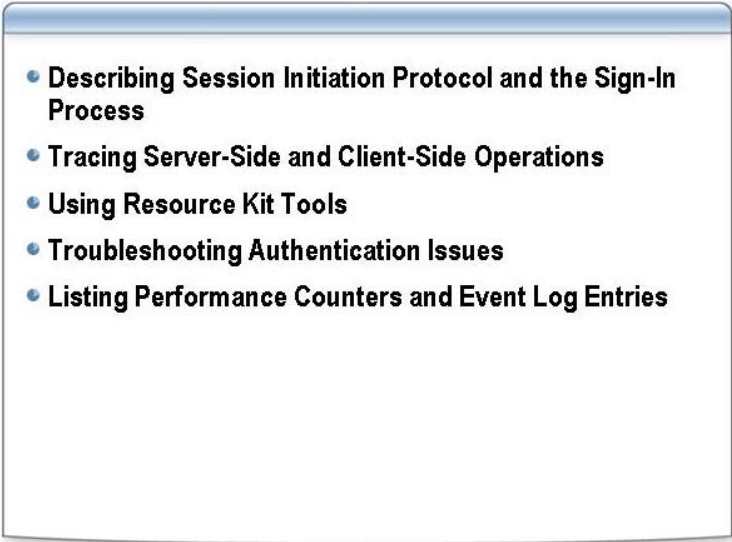Note that there may be multiple configuration errors on each computer.

Install the tools from the Live Communications Server 2005 Resource Kit to help you with troubleshooting. Exercise 1 Step 2 provides instructions for this installation process. Instructions for using these tools are in this module.

⚠ **Important: You have completed this lab successfully when Matt on 7034A-Client1-D and Holly on 7034A-FabrikamDC-D can view presence information and IM each other.**

Tasks	Detailed Steps
⚠ **Important:** You may have to log onto all of the VPC images to complete this lab.	
1. Log on to Communicator in Fabrikam and NWTraders.	a. Log on to **7034A-Client1-D** as **Administrator** with a password of **pass@word1**.
	b. Log on to Microsoft Office Communicator 2005 as **Matt@fabrikam.local** with a password of **pass@word1**.
	c. Switch to **7034A-NWTradersDC-D,** and log on as **NWTRADERS\Administrator** with a password of **pass@word1**.
	d. Log on to Microsoft Office Communicator 2005 as **Holly@nwtraders.local** with a password of **pass@word1**.
	e. Attempt to start an IM conversation with Matt Dawson.
2. Install the Live Communications Server	a. Click **Start**, and click **My Computer**.
	b. Browse to **E:\Demo Files\ LCS 2005 Resource Kit**, and double-click

Tasks	Detailed Steps
2005 Resource Kit	**LcResKit.msi**.
	c. The Welcome page of the installation wizard appears. Click **Next**.
	d. On the **License Agreement** page, click **I accept the terms in the license agreement**, and click **Next**.
	e. In the **Customer Information** page, accept the default settings, and click **Next**.
	f. In the **Custom Setup** page, accept the default installation location, and click **Next**.
	g. In the **Ready to Install the Program** page, click **Install**.
	h. Click **Finish** to exit the installation wizard.
	i. The tools are now in **C:\Program Files\Microsoft LC 2005\Reskit**. There are no Start Menu shortcuts for the tools.
3. Close down the Virtual PCs	a. When you have successfully completed the lab, in the virtual PC window for **7034A-LCSEESRV-D**, click the **Action** menu, and then click **Close**.
	b. In the **Close** dialog box, select **Turn off and delete changes**, then click **OK**.
	c. Repeat these steps on the remaining Virtual PCs.

Review

- **Describing Session Initiation Protocol and the Sign-In Process**
- **Tracing Server-Side and Client-Side Operations**
- **Using Resource Kit Tools**
- **Troubleshooting Authentication Issues**
- **Listing Performance Counters and Event Log Entries**

In this module, you looked at the tools and techniques to help you troubleshoot LCS 2005 with SP1. You started by looking at Session Initiation Protocol (SIP), which is the basis of instant messaging (IM). You then built on this knowledge to review numerous troubleshooting tools and methods, including use of the LCS 2005 with SP1 Resource Kit and the Capacity Planning Toolkit. You will now finish off this module (and the course) with a practical example of troubleshooting, where you attempt to use your knowledge to diagnose and fix an inoperative LCS 2005 with SP1 environment.