

z/OS Communications Server



# APPC Application Suite Administration

*Version 1 Release 2*



z/OS Communications Server



# APPC Application Suite Administration

*Version 1 Release 2*

**Note:**

Before using this information and the product it supports, be sure to read the general information under "Appendix E. Notices" on page 93.

**First Edition (October 2001)**

This edition applies to Version 1 Release 2 of z/OS (program number 5694-A01) and to all subsequent releases and modifications until otherwise indicated in new editions.

Publications are not stocked at the address given below. If you want more IBM publications, ask your IBM representative or write to the IBM branch office serving your locality.

Address comments to:

IBM Corporation  
Software Reengineering  
Department G71A/ Bldg 503  
Research Triangle Park, North Carolina 27709-9990  
U.S.A.

If you prefer to send comments electronically, use one of the following methods:

**Fax (USA and Canada):**

1-800-227-5088

**Internet e-mail:**

usib2hpd@vnet.ibm.com

**World Wide Web:**

<http://www.ibm.com/servers/eserver/zseries/zos>

**IBMLink:**

CIBMORCF at RALVM17

**IBM Mail Exchange:**

tkinlaw@us.ibm.com

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 1994, 2001. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>Figures</b> . . . . .	vii
<b>About This Book</b> . . . . .	ix
Where to Find More Information . . . . .	ix
Where to Find Related Information on the Internet . . . . .	ix
Licensed Documents . . . . .	x
LookAt, an Online Message Help Facility . . . . .	x
How to Contact IBM® Service . . . . .	xi
z/OS Communications Server Information . . . . .	xi
<b>Summary of Changes</b> . . . . .	xix
<b>Chapter 1. Overview</b> . . . . .	1
Product Overview . . . . .	1
APPC Connectivity Tester (APING) . . . . .	1
APPC File Transfer Protocol . . . . .	2
APPC NameServer . . . . .	2
A3270 Emulator Server . . . . .	2
<b>Chapter 2. Configuring the APPC Application Suite on VM</b> . . . . .	3
Before You Start . . . . .	3
Configuring AFTP Clients and Servers . . . . .	3
Task 1 - Locate the AFTP Files . . . . .	3
Task 2 - Add Side Information for the AFTP Client . . . . .	4
Task 3 - Add a Transaction Program Definition for the AFTP Server . . . . .	4
Task 4 - Update the Logon Profile . . . . .	5
Configuring APING Clients and Servers . . . . .	5
Task 1 - Locate the APING Files . . . . .	5
Task 2 - Add Side Information for the APING Client . . . . .	6
Task 3 - Add a Transaction Program Definition for the APING Server . . . . .	6
Task 4 - Update the Logon Profile . . . . .	7
Configuring the ANAME Client . . . . .	7
Task 1 - Locate the ANAME Client REXX EXEC . . . . .	7
Task 2 - Add Side Information for the ANAME Client . . . . .	7
Configuring the ANAME Server (ANAMED) . . . . .	8
Task 1 - Locate the ANAME Server Files . . . . .	8
Task 2 - Define the ANAME Server's Virtual Machine . . . . .	8
Configuring the A3270 Server . . . . .	10
Task 1 - Locate the A3270 Server Files . . . . .	10
Task 2 - Define the A3270 Server's Virtual Machine . . . . .	11
<b>Chapter 3. Configuring the APPC Application Suite on MVS</b> . . . . .	13
Before You Start . . . . .	13
Installing APPC Application Suite Messages . . . . .	13
Configuring AFTP Clients and Servers . . . . .	15
Task 1 - Modify the AFTP Client REXX EXEC . . . . .	15
Task 2 - Modify the ACOPY Client REXX EXEC . . . . .	16
Task 3 - Modify the AFTP Server REXX EXEC . . . . .	16
Task 4 - Add Side Information . . . . .	18
Task 5 - Add a Transaction Program Profile for the AFTP Server . . . . .	19
Configuring APING Clients and Servers . . . . .	20
Task 1 - Modify the APING Client REXX EXEC . . . . .	20
Task 2 - Add Side Information . . . . .	21

Task 3 - Add a Transaction Program (TP) Profile . . . . .	22
Configuring ANAME Clients . . . . .	24
Task 1 - Modify the ANAME Client REXX EXEC. . . . .	24
Task 2 - Add Side Information . . . . .	25
Configuring the ANAME Server . . . . .	26
Task 1 - Add a Transaction Program (TP) Profile . . . . .	26
Configuring the A3270 Server . . . . .	29
Task 1 - Define the A3270 Server to VTAM . . . . .	29
Task 2 - Configure VTAM to Start the A3270 Server Automatically . . . . .	33
Task 3 - Start the A3270 Application Program Major Node . . . . .	33
Task 4 - Modify the A3270 Server MVS PROC . . . . .	33
<b>Chapter 4. Setting Up Initialization Files.</b> . . . . .	35
The AFTP Initialization File . . . . .	35
Syntax . . . . .	35
Setting Up File Transfer Defaults . . . . .	35
Examples . . . . .	36
AFTP Initialization File Sample . . . . .	37
The ANAME Initialization File. . . . .	38
Syntax . . . . .	38
Defining ANAME System Administrators. . . . .	38
Preventing Updates . . . . .	39
Setting Trace for the ANAME Server . . . . .	40
ANAME Initialization File Sample . . . . .	40
The A3270 Initialization Files. . . . .	41
The VM A3270 Initialization File. . . . .	41
The MVS A3270 Initialization File . . . . .	42
<b>Chapter 5. Managing AFTP Servers</b> . . . . .	47
<b>Chapter 6. Managing the ANAME Server</b> . . . . .	51
Starting and Stopping the ANAME Server . . . . .	51
Starting the Server on VM. . . . .	51
Stopping the Server on VM . . . . .	52
Starting the Server on MVS . . . . .	52
Stopping the Server on MVS. . . . .	52
ANAME Database Administration . . . . .	52
Defining ANAME Administrators. . . . .	53
Managing Records for Other Users . . . . .	54
Adding Duplicate User Names . . . . .	54
Backing Up the ANAME Database. . . . .	55
Recovering the ANAME Database . . . . .	55
ANAME Administrator's Command Reference . . . . .	56
aname delete . . . . .	57
aname register . . . . .	59
anamed . . . . .	61
halt . . . . .	62
<b>Chapter 7. Managing the A3270 Emulator Server</b> . . . . .	63
How A3270 Works . . . . .	63
Using the A3270 Server in Your Network . . . . .	64
A Single Local Host System . . . . .	64
Multiple Local Host Systems . . . . .	64
Starting and Stopping the A3270 Server . . . . .	65
Starting the Server on VM. . . . .	66
Stopping the Server on VM . . . . .	66

Starting the Server on MVS . . . . .	66
Stopping the Server on MVS . . . . .	66
A3270 Server Command Reference . . . . .	66
a3270 . . . . .	68
display . . . . .	69
halt . . . . .	71
help . . . . .	72
selftest . . . . .	73
set . . . . .	74
terminate . . . . .	76
<b>Appendix A. How to Read a Syntax Diagram . . . . .</b>	<b>79</b>
Symbols and Punctuation . . . . .	79
Parameters . . . . .	79
Syntax Examples . . . . .	79
<b>Appendix B. A3270 Default Logmodes . . . . .</b>	<b>83</b>
<b>Appendix C. Configuring APPC to Support APPC Security SAME . . . . .</b>	<b>85</b>
What is APPC security=SAME? . . . . .	85
How is security=SAME Helpful to Users? . . . . .	85
How is APPC security=SAME Implemented? . . . . .	85
Network Configuration . . . . .	87
APPC/MVS Parameters . . . . .	87
Parameters for APPC/VM . . . . .	88
Recommendations for APPC/MVS . . . . .	88
Recommendations for APPC/VM . . . . .	89
<b>Appendix D. Information Apars . . . . .</b>	<b>91</b>
IP Information Apars . . . . .	91
SNA Information Apars . . . . .	92
<b>Appendix E. Notices . . . . .</b>	<b>93</b>
Trademarks . . . . .	96
<b>Index . . . . .</b>	<b>99</b>





---

## Figures

1.	Sample Symbolic Destination Name Definition for AFTP . . . . .	4
2.	Sample \$SERVER\$ NAMES File . . . . .	5
3.	Sample Symbolic Destination Name Definition for APING . . . . .	6
4.	Sample \$SERVER\$ NAMES File . . . . .	7
5.	Sample Symbolic Destination Name Definition for ANAME . . . . .	8
6.	Sample ANAME Virtual Machine CP Directory . . . . .	9
7.	Sample A3270 VM/ESA Virtual Machine Directory. . . . .	12
8.	Partial Sample of the AFTP EXEC . . . . .	15
9.	Partial Sample of the ACOPY EXEC . . . . .	16
10.	Partial Sample of the AFTPD EXEC . . . . .	17
11.	Sample AFTP Side Information JCL . . . . .	18
12.	Partial Sample AFTPD TP Add JCL . . . . .	20
13.	Sample APING EXEC . . . . .	21
14.	Sample APING Add Side Information JCL. . . . .	22
15.	Partial Sample APINGD Add TP Profile JCL . . . . .	23
16.	Sample ANAME EXEC . . . . .	25
17.	Sample ANAME Side Information JCL . . . . .	26
18.	Partial Sample ANAME Add TP Profile JCL . . . . .	28
19.	Sample A3270 VTAM Application Program Major Node Definition . . . . .	32
20.	Sample A3270 MVS PROC . . . . .	34
21.	Sample AFTP Initialization File . . . . .	37
22.	Sample ANAME Initialization File . . . . .	41
23.	Sample A3270 Virtual Machine Profile . . . . .	42
24.	Sample A3270 Initialization File . . . . .	43
25.	Sample AFTP Server Audit Log . . . . .	48
26.	Simple VM Example . . . . .	64
27.	Simple MVS Example . . . . .	64
28.	A Complex Network . . . . .	65



---

## About This Book

This book describes the primary tasks required to configure the IBM APPC Application Suite on the MVS and VM operating systems, and to manage the APPC NameServer (ANAME) and A3270 emulator servers.

This book is for:

- System programmers or network administrators who are responsible for configuring the APPC Application Suite on either the VM or MVS operating systems, and for managing the ANAME and A3270 servers. You should know how to perform basic APPC configuration tasks for the operating system platform you are using.
- VM end users who want to create side information for the APPC Application Suite clients.
- APPC File Transfer Protocol (AFTP) end users who want to create a personal copy of the AFTP initialization file.

---

## Where to Find More Information

This section contains:

- Pointers to information available on the Internet
- Information about licensed documentation
- Information about LookAt, the online message tool
- A set of tables that describes the books in the z/OS Communications Server (z/OS CS) library, along with related publications

## Where to Find Related Information on the Internet

Home Page	Web address
z/OS	<a href="http://www.ibm.com/servers/eserver/zseries/zos/">http://www.ibm.com/servers/eserver/zseries/zos/</a>
z/OS Internet Library	<a href="http://www.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www.ibm.com/servers/eserver/zseries/zos/bkserv/</a>
IBM Communications Server product	<a href="http://www.software.ibm.com/network/commserver/">http://www.software.ibm.com/network/commserver/</a>
IBM Communications Server support	<a href="http://www.software.ibm.com/network/commserver/support/">http://www.software.ibm.com/network/commserver/support/</a>
IBM Systems Center publications	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>
IBM Systems Center flashes	<a href="http://www-1.ibm.com/support/techdocs/atsmastr.nsf">http://www-1.ibm.com/support/techdocs/atsmastr.nsf</a>
VTAM and TCP/IP	<a href="http://www.software.ibm.com/network/commserver/about/csos390.html">http://www.software.ibm.com/network/commserver/about/csos390.html</a>
IBM	<a href="http://www.ibm.com">http://www.ibm.com</a>
RFC	<a href="http://www.ietf.org/rfc.html">http://www.ietf.org/rfc.html</a>

Information about Web addresses can also be found in informational APAR II11334.

### DNS Web Sites

For information about DNS, see the following Web sites:

#### USENET news groups:

[comp.protocols.dns.bind](mailto:comp.protocols.dns.bind)

For BIND mailing lists, see:

- <http://www.isc.org/ml-archives/>
  - BIND Users
    - Subscribe by sending mail to [bind-users-request@isc.org](mailto:bind-users-request@isc.org)
    - Submit questions or answers to this forum by sending mail to [bind-users@isc.org](mailto:bind-users@isc.org)
  - BIND 9 Users (Note: This list may not be maintained indefinitely.)
    - Subscribe by sending mail to [bind9-users-request@isc.org](mailto:bind9-users-request@isc.org)
    - Submit questions or answers to this forum by sending mail to [bind9-users@isc.org](mailto:bind9-users@isc.org)

For definitions of the terms and abbreviations used in this book, you can view or download the latest *IBM Glossary of Computing Terms* at the following Web address:

<http://www.ibm.com/ibm/terminology>

**Note:** Any pointers in this publication to Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

## Licensed Documents

z/OS Communications Server licensed documentation in PDF format is available on the Internet at the IBM Resource Link Web site at <http://www.ibm.com/servers/resourcelink>. Licensed books are available only to customers with a z/OS Communications Server license. Access to these books requires an IBM Resource Link Web user ID and password, and a key code. With your z/OS Communications Server order, you received a memo that includes this key code. To obtain your IBM Resource Link Web user ID and password, log on to <http://www.ibm.com/servers/resourcelink>. To register for access to the z/OS licensed books perform the following steps:

1. Log on to Resource Link using your Resource Link user ID and password.
2. Click on **User Profiles** located on the left-hand navigation bar.
3. Click on **Access Profile**.
4. Click on **Request Access to Licensed books**.
5. Supply your key code where requested and click on the **Submit** button.

If you supplied the correct key code, you will receive confirmation that your request is being processed. After your request is processed, you will receive an e-mail confirmation.

You cannot access the z/OS licensed books unless you have registered for access to them and received an e-mail confirmation informing you that your request has been processed. To access the licensed books:

1. Log on to Resource Link using your Resource Link user ID and password.
2. Click on **Library**.
3. Click on **zSeries**.
4. Click on **Software**.
5. Click on **z/OS Communications Server**.
6. Access the licensed book by selecting the appropriate element.

## LookAt, an Online Message Help Facility

LookAt is an online facility that allows you to look up explanations for z/OS CS messages and system abends.

Using LookAt to find information is faster than a conventional search because LookAt goes directly to the explanation.

LookAt can be accessed from the Internet or from a TSO command line.

To use LookAt as a TSO command, LookAt must be installed on your host system. You can obtain the LookAt code for TSO from the LookAt Web site by clicking on **News and Help** or from the z/OS V1R2 Collection, SK3T-4269.

To find a message explanation from a TSO command line, simply enter **lookat+message ID**, as in the following example:

```
lookat ezz8477i
```

This results in direct access to the message explanation for message EZZ8477I.

You can use LookAt on the Internet at the following Web site:  
[www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/lookat.html](http://www.ibm.com/servers/eserver/zseries/zos/bkserv/lookat/lookat.html)

To find a message explanation from the LookAt Web site, simply enter the message ID. You can select the release, if applicable.

## How to Contact IBM® Service

For telephone assistance in problem diagnosis and resolution (in the United States or Puerto Rico), call the IBM Software Support Center anytime (1-800-237-5511). You will receive a return call within 8 business hours (Monday – Friday, 8:00 a.m. – 5:00 p.m., local customer time).

Outside of the United States or Puerto Rico, contact your local IBM representative or your authorized IBM supplier.

## z/OS Communications Server Information

This section contains descriptions of the books in the z/OS Communications Server library.

z/OS Communications Server publications are available:

- Online at the z/OS Internet Library web page at <http://www.ibm.com/servers/eserver/zseries/zos/>
- In hardcopy and softcopy
- In softcopy only

### Softcopy Information

Softcopy publications are available in the following collections:

Titles	Order Number	Description
<i>z/OS V1R2 Collection</i>	SK3T-4269	This is the CD collection shipped with the z/OS product. It includes the libraries for z/OS V1R2, in both BookManager and PDF formats.
<i>z/OS Software Products Collection</i>	SK3T-4270	This CD includes, in both BookManager and PDF formats, the libraries of z/OS software products that run on z/OS but are not elements and features, as well as the <i>Getting Started with Parallel Sysplex</i> bookshelf.

<b>Titles</b>	<b>Order Number</b>	<b>Description</b>
<i>z/OS V1R2 and Software Products DVD Collection</i>	SK3T-4271	This collection includes the libraries of z/OS (the element and feature libraries) and the libraries for z/OS software products in both BookManager and PDF format. This collection combines SK3T-4269 and SK3T-4270.
<i>z/OS Licensed Product Library</i>	SK3T-4307	This CD includes the licensed books in both BookManager and PDF format.
<i>System Center Publication IBM S/390 Redbooks Collection</i>	SK2T-2177	This collection contains over 300 ITSO redbooks that apply to the S/390 platform and to host networking arranged into subject bookshelves.

## **z/OS Communications Server Library**

The following abbreviations follow each order number in the tables below.

**HC/SC** — Both hardcopy and softcopy are available.

**SC** — Only softcopy is available. These books are available on the CD Rom accompanying z/OS (SK3T-4269 or SK3T-4307). Unlicensed books can be viewed at the z/OS Internet library site.

Updates to books are available on RETAIN and in the document called *OS/390 DOC APARs and ++HOLD DOC data* which can be found at [http://www.s390.ibm.com/os390/bkserv/new\\_tech\\_info.html](http://www.s390.ibm.com/os390/bkserv/new_tech_info.html). See “Appendix D. Information Apars” on page 91 for a list of the books and the informational apars (info apars) associated with them.

### ***Planning and Migration:***

<b>Title</b>	<b>Number</b>	<b>Format</b>	<b>Description</b>
<i>z/OS Communications Server: SNA Migration</i>	GC31-8774	HC/SC	This book is intended to help you plan for SNA, whether you are migrating from a previous version or installing SNA for the first time. This book also identifies the optional and required modifications needed to enable you to use the enhanced functions provided with SNA.
<i>z/OS Communications Server: IP Migration</i>	GC31-8773	HC/SC	This book is intended to help you plan for TCP/IP Services, whether you are migrating from a previous version or installing IP for the first time. This book also identifies the optional and required modifications needed to enable you to use the enhanced functions provided with TCP/IP Services.

### ***Resource Definition, Configuration, and Tuning:***

<b>Title</b>	<b>Number</b>	<b>Format</b>	<b>Description</b>
<i>z/OS Communications Server: IP Configuration Guide</i>	SC31-8775	HC/SC	This book describes the major concepts involved in understanding and configuring an IP network. Familiarity with the z/OS operating system, IP protocols, z/OS UNIX System Services, and IBM Time Sharing Option (TSO) is recommended. Use this book in conjunction with the <i>z/OS Communications Server: IP Configuration Reference</i> .

<b>Title</b>	<b>Number</b>	<b>Format</b>	<b>Description</b>
<i>z/OS Communications Server: IP Configuration Reference</i>	SC31-8776	HC/SC	This book presents information for people who want to administer and maintain IP. Use this book in conjunction with the <i>z/OS Communications Server: IP Configuration Guide</i> . The information in this book includes: <ul style="list-style-type: none"> <li>• TCP/IP configuration data sets</li> <li>• Configuration statements</li> <li>• Translation tables</li> <li>• SMF records</li> <li>• Protocol number and port assignments</li> </ul>
<i>z/OS Communications Server: SNA Network Implementation Guide</i>	SC31-8777	HC/SC	This book presents the major concepts involved in implementing an SNA network. Use this book in conjunction with the <i>z/OS Communications Server: SNA Resource Definition Reference</i> .
<i>z/OS Communications Server: SNA Resource Definition Reference</i>	SC31-8778	HC/SC	This book describes each SNA definition statement, start option, and macroinstruction for user tables. It also describes NCP definition statements that affect SNA. Use this book in conjunction with the <i>z/OS Communications Server: SNA Network Implementation Guide</i> .
<i>z/OS Communications Server: SNA Resource Definition Samples</i>	SC31-8836	SC	This book contains sample definitions to help you implement SNA functions in your networks, and includes sample major node definitions.
<i>z/OS Communications Server: AnyNet SNA over TCP/IP</i>	SC31-8832	SC	This guide provides information to help you install, configure, use, and diagnose SNA over TCP/IP.
<i>z/OS Communications: Server AnyNet Sockets over SNA</i>	SC31-8831	SC	This guide provides information to help you install, configure, use, and diagnose sockets over SNA. It also provides information to help you prepare application programs to use sockets over SNA.

### **Operation:**

<b>Title</b>	<b>Number</b>	<b>Format</b>	<b>Description</b>
<i>z/OS Communications Server: IP User's Guide and Commands</i>	SC31-8780	HC/SC	This book describes how to use TCP/IP applications. It contains requests that allow a user to: log on to a remote host using Telnet, transfer data sets using FTP, send and receive electronic mail, print on remote printers, and authenticate network users.
<i>z/OS Communications Server: IP System Administrator's Commands</i>	SC31-8781	HC/SC	This book describes the functions and commands helpful in configuring or monitoring your system. It contains system administrator's commands, such as NETSTAT, PING, TRACERTE and their UNIX counterparts. It also includes TSO and MVS commands commonly used during the IP configuration process.
<i>z/OS Communications Server: SNA Operation</i>	SC31-8779	HC/SC	This book serves as a reference for programmers and operators requiring detailed information about specific operator commands.
<i>z/OS Communications Server: Operations Quick Reference</i>	SX75-0124	HC/SC	This book contains essential information about SNA and IP commands.

### **Customization:**

Title	Number	Format	Description
<i>z/OS Communications Server: SNA Customization</i>	LY43-0092	SC	This book enables you to customize SNA, and includes the following: <ul style="list-style-type: none"> <li>• Communication network management (CNM) routing table</li> <li>• Logon-interpret routine requirements</li> <li>• Logon manager installation-wide exit routine for the CLU search exit</li> <li>• TSO/SNA installation-wide exit routines</li> <li>• SNA installation-wide exit routines</li> </ul>
<i>z/OS Communications Server: IP Network Print Facility</i>	SC31-8833	SC	This book is for system programmers and network administrators who need to prepare their network to route SNA, JES2, or JES3 printer output to remote printers using TCP/IP Services.

### **Writing Application Programs:**

Title	Number	Format	Description
<i>z/OS Communications Server: IP Application Programming Interface Guide</i>	SC31-8788	SC	This book describes the syntax and semantics of program source code necessary to write your own application programming interface (API) into TCP/IP. You can use this interface as the communication base for writing your own client or server application. You can also use this book to adapt your existing applications to communicate with each other using sockets over TCP/IP.
<i>z/OS Communications Server: IP CICS Sockets Guide</i>	SC31-8807	SC	This book is for people who want to set up, write application programs for, and diagnose problems with the socket interface for CICS using z/OS TCP/IP.
<i>z/OS Communications Server: IP IMS Sockets Guide</i>	SC31-8830	SC	This book is for programmers who want application programs that use the IMS TCP/IP application development services provided by IBM's TCP/IP Services.
<i>z/OS Communications Server: IP Programmer's Reference</i>	SC31-8787	SC	This book describes the syntax and semantics of a set of high-level application functions that you can use to program your own applications in a TCP/IP environment. These functions provide support for application facilities, such as user authentication, distributed databases, distributed processing, network management, and device sharing. Familiarity with the z/OS operating system, TCP/IP protocols, and IBM Time Sharing Option (TSO) is recommended.
<i>z/OS Communications Server: SNA Programming</i>	SC31-8829	SC	This book describes how to use SNA macroinstructions to send data to and receive data from (1) a terminal in either the same or a different domain, or (2) another application program in either the same or a different domain.
<i>z/OS Communications Server: SNA Programmers LU 6.2 Guide</i>	SC31-8811	SC	This book describes how to use the SNA LU 6.2 application programming interface for host application programs. This book applies to programs that use only LU 6.2 sessions or that use LU 6.2 sessions along with other session types. (Only LU 6.2 sessions are covered in this book.)



Title	Number	Format	Description
<i>z/OS Communications Server: SNA Programmers LU 6.2 Reference</i>	SC31-8810	SC	This book provides reference material for the SNA LU 6.2 programming interface for host application programs.
<i>z/OS Communications Server: CSM Guide</i>	SC31-8808	SC	This book describes how applications use the communications storage manager.
<i>z/OS Communications Server: CMIP Services and Topology Agent Guide</i>	SC31-8828	SC	This book describes the Common Management Information Protocol (CMIP) programming interface for application programmers to use in coding CMIP application programs. The book provides guide and reference information about CMIP services and the SNA topology agent.

### **Diagnosis:**

Title	Number	Format	Description
<i>z/OS Communications Server: IP Diagnosis</i>	GC31-8782	HC/SC	This book explains how to diagnose TCP/IP problems and how to determine whether a specific problem is in the TCP/IP product code. It explains how to gather information for and describe problems to the IBM Software Support Center.
<i>z/OS Communications Server: SNA Diagnosis V1 Techniques and Procedures</i> and <i>z/OS Communications Server: SNA Diagnosis V2 FFST Dumps and the VIT</i>	LY43-0088 LY43-0089	HC/SC	These books help you identify an SNA problem, classify it, and collect information about it before you call the IBM Support Center. The information collected includes traces, dumps, and other problem documentation.
<i>z/OS Communications Server: SNA Data Areas Volume 1</i> and <i>z/OS Communications Server: SNA Data Areas Volume 2</i>	LY43-0090 LY43-0091	SC	These books describe SNA data areas and can be used to read an SNA dump. They are intended for IBM programming service representatives and customer personnel who are diagnosing problems with SNA.

### **Messages and Codes:**

Title	Number	Format	Description
<i>z/OS Communications Server: SNA Messages</i>	SC31-8790	HC/SC	This book describes the ELM, IKT, IST, ISU, IUT, IVT, and USS messages. Other information in this book includes: <ul style="list-style-type: none"> <li>• Command and RU types in SNA messages</li> <li>• Node and ID types in SNA messages</li> <li>• Supplemental message-related information</li> </ul>
<i>z/OS Communications Server: IP Messages Volume 1 (EZA)</i>	SC31-8783	HC/SC	This volume contains TCP/IP messages beginning with EZA.
<i>z/OS Communications Server: IP Messages Volume 2 (EZB)</i>	SC31-8784	HC/SC	This volume contains TCP/IP messages beginning with EZB.
<i>z/OS Communications Server: IP Messages Volume 3 (EZY)</i>	SC31-8785	HC/SC	This volume contains TCP/IP messages beginning with EZY.

Title	Number	Format	Description
<i>z/OS Communications Server: IP Messages Volume 4 (EZZ-SNM)</i>	SC31-8786	HC/SC	This volume contains TCP/IP messages beginning with EZZ and SNM.
<i>z/OS Communications Server: IP and SNA Codes</i>	SC31-8791	HC/SC	This book describes codes and other information that appear in z/OS Communications Server messages.

### **APPC Application Suite:**

Title	Number	Format	Description
<i>z/OS Communications Server: APPC Application Suite User's Guide</i>	GC31-8809	SC	This book documents the end-user interface (concepts, commands, and messages) for the AFTP, ANAME, and APING facilities of the APPC application suite. Although its primary audience is the end user, administrators and application programmers may also find it useful.
<i>z/OS Communications Server APPC Application Suite Administration</i>	SC31-8835	SC	This book contains the information that administrators need to configure the APPC application suite and to manage the APING, ANAME, AFTP, and A3270 servers.
<i>z/OS Communications Server: APPC Application Suite Programming</i>	SC31-8834	SC	This book provides the information application programmers need to add the functions of the AFTP and ANAME APIs to their application programs.

### **Redbooks**

The following Redbooks may help you as you implement z/OS Communications Server.

Title	Number
<i>TCP/IP Tutorial and Technical Overview</i>	GG24-3376
<i>SNA and TCP/IP Integration</i>	SG24-5291
<i>IBM Communication Server for OS/390 V2R10 TCP/IP Implementation Guide: Volume 1: Configuration and Routing</i>	SG24-5227
<i>IBM Communication Server for OS/390 V2R10 TCP/IP Implementation Guide: Volume 2: UNIX Applications</i>	SG24-5228
<i>IBM Communication Server for OS/390 V2R10 TCP/IP Implementation Guide: Volume 3: MVS Applications</i>	SG24-5229
<i>OS/390 Secureway Communication Server V2R8 TCP/IP Guide to Enhancements</i>	SG24-5631
<i>TCP/IP in a Sysplex</i>	SG24-5235
<i>Managing OS/390 TCP/IP with SNMP</i>	SG24-5866
<i>Security in OS/390-based TCP/IP Networks</i>	SG24-5383
<i>IP Network Design Guide</i>	SG24-2580

### **Related Information**

For information about z/OS products, refer to *z/OS Information Roadmap* (SA22-7500). The Roadmap describes what level of documents are supplied with each release of z/OS Communications Server, as well as describing each z/OS publication.

The table below lists books that may be helpful to readers.

<b>Title</b>	<b>Number</b>
<i>z/OS SecureWay Security Server Firewall Technologies</i>	SC24-5922
<i>S/390: OSA-Express Customer's Guide and Reference</i>	SA22-7403
<i>z/OS MVS Diagnosis: Procedures</i>	GA22-7587
<i>z/OS MVS Diagnosis: Reference</i>	GA22-7588
<i>z/OS MVS Diagnosis: Tools and Service Aids</i>	GA22-7589

### **Determining If a Publication Is Current**

As needed, IBM updates its publications with new and changed information. For a given publication, updates to the hardcopy and associated BookManager softcopy are usually available at the same time. Sometimes, however, the updates to hardcopy and softcopy are available at different times. Here is how to determine if you are looking at the most current copy of a publication:

1. At the end of a publication's order number there is a dash followed by two digits, often referred to as the dash level. A publication with a higher dash level is more current than one with a lower dash level. For example, in the publication order number GC28-1747-07, the dash level 07 means that the publication is more current than previous levels, such as 05 or 04.
2. If a hardcopy publication and a softcopy publication have the same dash level, it is possible that the softcopy publication is more current than the hardcopy publication. Check the dates shown in the Summary of Changes. The softcopy publication might have a more recently dated Summary of Changes than the hardcopy publication.
3. To compare softcopy publications, you can check the last two characters of the publication's filename (also called the book name). The higher the number, the more recent the publication. Also, next to the publication titles in the CD-ROM booklet and the readme files, there is an asterisk (\*) that indicates whether a publication is new or changed.



---

## Summary of Changes

|                   **Summary of Changes**  
|                   **for SC31-8835-00**  
|                   **z/OS Version 1 Release 2**

|                   The book contains information previously presented in *OS/390 V2R5 eNetwork*  
|                   *Communications Server: APPC Applications Suite Administration*, SC31-8620.

|                   This book contains minor terminology, maintenance, and editorial changes.  
|                   Technical changes or additions to the text and illustrations are indicated by a  
|                   vertical line to the left of the change.



---

## Chapter 1. Overview

The IBM\* APPC Application Suite is a set of common applications designed to enhance the value of SNA networks for end users. These applications simplify the use of SNA networks, providing solutions for common end-user tasks such as file transfer, terminal emulation, connectivity testing, and name registration. The APPC Application Suite also includes application programming interfaces (APIs) for use by programmers, and special services to assist administrators in making SNA networks easier to use.

These applications can be used in both traditional SNA networks and Advanced Peer-to-Peer Networking\* (APPN\*) networks. Designed to promote multivendor connectivity, the APPC Application Suite functions are intended to work with APPC Application Suite implementations on other IBM and non-IBM operating environments as those offerings become available.

---

### Product Overview

The APPC Application Suite is a set of client/server applications that run on advanced program-to-program (APPC) sessions. Some of the applications provide an application programming interface (API) so that additional applications can be developed which access these features and functions.

The following parts are shipped in the IBM APPC Application Suite for MVS and VM:

Facility	Client	Server	API
APPC Connectivity Tester (APING)	X	X	
APPC File Transfer Protocol (AFTP)	X	X	X
APPC NameServer (ANAME)	X	X	X
APPC 3270 Emulator Server (A3270)		X	

Some of the APPC Application Suite clients and servers are available on other operating systems, enabling communication across platforms. For example, the APPC Application Suite is shipped with the productivity aids of Communications Manager Version 1.11. SNA Desktop/6000 V1.1 provides the APPC APPC Application Suite functions on the AIX\* platform, including an A3270 client application.

### APPC Connectivity Tester (APING)

APING is an application that verifies network connections. It is very useful for end users, system administrators, and support personnel in establishing network connectivity, problem determination, and simple performance measurements. APING's operation and its command line interface are modeled after similar programs widely used in other networks. APING should be the first program you run after configuring APPC to test the new configuration. APING returns detailed error information if problems are detected.

APING provides the following functions for APPC verification:

- Determines whether a session can be set up between two computers.
- Exchanges data packets.
- Times the length of time for a data transfer.

- Displays error data if session allocation fails.

## **APPC File Transfer Protocol**

The APPC File Transfer Protocol (AFTP) facility provides file manipulation and transfer services to application programs and end users. With AFTP, you can copy text and binary files between your computer and any computer running the AFTP server.

The AFTP facility also has commands that help you locate the files you want to work with and allows you to:

- Set and query the current directory.
- List files in a selected directory.

## **APPC NameServer**

The APPC NameServer (ANAME) facility reduces some of the complexity of APPC by simplifying network naming. It allows users to register nicknames associated with their fully qualified LU names in a globally accessible directory. It also allows you to associate multiple LUs with a group and to identify transaction programs available at an LU. All of the clients and servers in the APPC Application Suite are able to access the information in the ANAME database.

## **A3270 Emulator Server**

The A3270 server allows users on workstations to establish a 3270 emulator connection to the host using APPC communications. The A3270 server function does not have a user interface. The user interface is provided by the client code on the workstation and is documented in the client documentation. SNA Desktop/6000 on the AIX platform includes an A3270 client application.



---

## Chapter 2. Configuring the APPC Application Suite on VM

This chapter describes the configuration tasks required to define the APPC Application Suite to APPC/VM VTAM Support (AVS) on the VM operating system platform. The APPC Application Suite provides four applications: AFTP, APING, ANAME, and the A3270 server. See “Product Overview” on page 1 for a brief overview of each application. See *VM/ESA Connectivity Planning, Administration, and Operation* for additional information about AVS.

Because the configuration tasks for each application vary, configuration for each application is described in a separate section. For information about configuring the APPC Application Suite on MVS, see “Chapter 3. Configuring the APPC Application Suite on MVS” on page 13.

---

### Before You Start

Before you configure the APPC Application Suite, ensure that:

- The required system software is installed.
- The APPC Application Suite is installed.
- AVS is configured.

For information about system software requirements and installation, see the *Program Directory for APPC Application Suite - VM Version 1*.

---

### Configuring AFTP Clients and Servers

To configure AFTP on VM:

1. Locate the AFTP files.
2. Add side information.
3. Add transaction program definitions.
4. Update the logon profile.

#### Task 1 - Locate the AFTP Files

When the APPC Application Suite was installed, the AFTP files should have been copied and renamed.

Table 1 contains a description of each AFTP file, the name of the file that was retrieved from the IBM Software Distribution tape (*installed file name*) and the name of the file that is used by the APPC Application Suite programs (*production file name*). Each AFTP file must be renamed *from* the installed file name *to* the production file name.

If the AFTP files have not been renamed, rename the files before continuing to the next step. See the *Program Directory for APPC Application Suite, VM Version 1* for additional information about copying and renaming the AFTP files.

Table 1. AFTP Files

File Description	Installed File Name	Production File Name
AFTP client EXEC	APPFAFVX EXEC	AFTP EXEC
ACOPY client EXEC	APPFACVX EXEC	ACOPY EXEC
AFTP server EXEC	APPFADVX EXEC	AFTPD EXEC
AFTP initialization file	APPFAFVI INI	AFTP INI

## Task 2 - Add Side Information for the AFTP Client

Because the VM operating system does not allow the use of explicit fully qualified LU names, a nickname must be defined for each AFTP server to which AFTP clients will connect. Nicknames can be defined using ANAME, or using APPC side information. Nicknames defined using ANAME are called *user names*. Nicknames defined using APPC side information are called *symbolic destination names*. On VM, end users must always use either user names or symbolic destination names when specifying destinations. For additional information about using ANAME, see the *z/OS Communications Server: APPC Application Suite User's Guide*. For additional information about side information, see the *Common Programming Interface Communications Reference*.

If you do not require symbolic destination names, you can skip this step and proceed to “Task 3 - Add a Transaction Program Definition for the AFTP Server”.

VM implements side information using CMS communications directory files. A communications directory file is a CMS NAMES format file that can be set up on a system level (by the system administrator) or on a user level. The default name for the system-level communications directory is SCOMDIR NAMES; the default name for the user-level file is UCOMDIR NAMES.

A sample SCOMDIR NAMES file is shown in Figure 1, which contains sample side information definitions for several AFTP servers.

```
:nick.symdestname1 :luname.gateway luname
:nick.symdestname2 :luname.gateway luname
:nick.symdestname3 :luname.gateway luname
```

Figure 1. Sample Symbolic Destination Name Definition for AFTP

### **:nick.**

Specifies a symbolic destination name for an AFTP server.

### **:luname.**

Specifies the location of an AFTP server. The LU name consists of two fields of up to eight characters each, separated by a space. The first field contains the AFTP client's gateway. The second field contains an AFTP server's LU name.

For a detailed description of the use and contents of the SCOMDIR or UCOMDIR files, see *VM/ESA Connectivity Planning, Administration, and Operation*.

## Task 3 - Add a Transaction Program Definition for the AFTP Server

A REXX EXEC starts the AFTP server (AFTPD). The default name of the REXX EXEC is AFTPD EXEC. If the name of the AFTPD EXEC has been changed, add a transaction program definition for the AFTP server to the \$SERVER\$ NAMES file. Otherwise, you can skip this task and proceed to “Task 4 - Update the Logon Profile” on page 5.

An example of a \$SERVER\$ NAMES file that contains a transaction program definition for AFTPD is shown in Figure 2 on page 5. In the example, *execname* is the new name of the REXX EXEC that starts AFTPD.

```

:nick.AFTPD          :list.*
                    :module.execname

```

Figure 2. Sample \$SERVER\$ NAMES File

For a detailed description of the use and contents of the \$SERVER\$ NAMES file, see *VM/ESA Connectivity Planning, Administration, and Operation*

## Task 4 - Update the Logon Profile

Add the following statements to the logon profile of each user who installs the AFTP server. If a logon profile already contains some or all of these statements, they do not need to be added again.

### SET SERVER ON

Enables CMS to start the AFTP transaction program.

**Note:** CMS cannot start the AFTP server if another program, such as FILELIST, is already active on your virtual machine.

### SET FULLSCREEN OFF

Ensures that you are not using fullscreen CMS. The AFTP server cannot start when fullscreen CMS is active.

### SET AUTOREAD OFF

Ensures that the virtual machine returns to the VM RUNNING state after a command is completed. The AFTP server can only start when VM is in the RUNNING state. (The server cannot start when VM is in the VM READ state.) When AUTOREAD is set OFF, the virtual machine returns to the RUNNING state after each command completes.

---

## Configuring APING Clients and Servers

To configure APING on VM:

1. Locate the APING files.
2. Add side information.
3. Add a transaction program definition.
4. Update the logon profile.

## Task 1 - Locate the APING Files

When the APPC Application Suite was installed, the APING files should have been copied and renamed. Table 2 contains a description of each APING file, the name of the file that was retrieved from the IBM Software Distribution tape (*installed file name*) and the name of the file that is used by the APPC Application Suite programs (*production file name*). Each APING file must be renamed *from* the installed file name *to* the production file name.

If the APING files have not been renamed, rename the files before continuing to the next step. See the *Program Directory for APPC Application Suite, VM Version 1* for additional information about copying and renaming the APING files.

Table 2. APING Files

File Description	Installed File Name	Production File Name
APING client REXX EXEC	APPPADVX EXEC	APING EXEC
APING server REXX EXEC	APPPAPVX EXEC	APINGD EXEC

## Task 2 - Add Side Information for the APING Client

Because the VM operating system does not allow the use of explicit fully qualified LU names, a nickname must be defined for each APING server to which APING clients will connect. Nicknames can be defined using ANAME, or using APPC side information. Nicknames defined using ANAME are called *user names*. Nicknames defined using APPC side information are called *symbolic destination names*. On VM, end users must always use either user names or symbolic destination names when specifying destinations. For additional information about using ANAME, see the *z/OS Communications Server: APPC Application Suite User's Guide*. For additional information about side information, see the *Common Programming Interface Communications Reference*.

If you do not require symbolic destination names, you can skip this step and proceed to “Task 3 - Add a Transaction Program Definition for the APING Server”.

VM implements side information using CMS communications directory files. A communications directory file is a CMS NAMES format file that can be set up on a system level (by the system administrator) or on a user level. The default name for the system-level communications directory is SCOMDIR NAMES; the default name for the user-level file is UCOMDIR NAMES.

A sample SCOMDIR NAMES file is shown in Figure 3, which contains sample side information definitions for several APING servers.

```
:nick.symdestname1 :luname.gateway luname
:nick.symdestname2 :luname.gateway luname
:nick.symdestname3 :luname.gateway luname
```

Figure 3. Sample Symbolic Destination Name Definition for APING

### **:nick.**

Specifies a symbolic destination name for an APING server.

### **:luname.**

Specifies the location of an APING server. The LU name consists of two fields of up to eight characters each, separated by a space. The first field contains the APING client's gateway. The second field contains an APING server's LU name.

For detailed information about the use and contents of SCOMDIR or UCOMDIR files, see *VM/ESA Connectivity Planning, Administration, and Operation*.

## Task 3 - Add a Transaction Program Definition for the APING Server

A REXX EXEC starts the APING server (APINGD). The default name of the REXX EXEC is APINGD EXEC. If the name of the APINGD EXEC has been changed, add a transaction program definition for the APING server to the \$SERVER\$ NAMES file. Otherwise, you can skip this task and go to Task 3.

An example of a \$SERVER\$ NAMES file that contains a transaction program definition for APINGD is shown in Figure 4 on page 7. In the example, *execname* is the new name of the REXX EXEC that starts APINGD.

```
:nick.APINGD          :list.*
                      :module.execname
```

Figure 4. Sample `$$SERVER$ NAMES` File

For a detailed description of the use and contents of the `$$SERVER$ NAMES` file, see *VM/ESA Connectivity Planning, Administration, and Operation*.

## Task 4 - Update the Logon Profile

Add the following statements to the logon profile of each user who installs the APING server. If the logon profile already contains some or all of these statements, they do not need to be added again.

### SET SERVER ON

Enables CMS to start the APINGD transaction program.

**Note:** CMS cannot start server transaction programs if another program, such as FILELIST, is already active on your virtual machine.

### SET FULLSCREEN OFF

Ensures that you are not using fullscreen CMS. The APING server cannot start when fullscreen CMS is active.

### SET AUTOREAD OFF

Ensures that the virtual machine returns to the VM RUNNING state after a command is completed. The APING server can only start when VM is in the RUNNING state. (The server cannot start when VM is in the VM READ state.) When AUTOREAD is set OFF, the virtual machine returns to the RUNNING state after each command completes.

---

## Configuring the ANAME Client

To configure the ANAME client on VM:

1. Locate the ANAME client REXX EXEC
2. Add side information.

## Task 1 - Locate the ANAME Client REXX EXEC

Before you configure the ANAME client, you must locate the ANAME EXEC file. When the APPC Application Suite was installed, the ANAME EXEC file was named APPMANVX EXEC. The file should have been copied and renamed to ANAME EXEC. If the file was not renamed, you need to rename the file before continuing to the next step.

## Task 2 - Add Side Information for the ANAME Client

Because the VM operating system does not allow the use of explicit fully qualified LU names, a symbolic destination name must be defined for the ANAME server using APPC side information

VM implements side information using CMS communications directory files. A communications directory file is a CMS NAMES format file that can be set up on a system level (by the system administrator) or on a user level. The default name for the system-level communications directory is SCOMDIR NAMES; the default name for the user-level file is UCOMDIR NAMES. For additional information about side information, see the *Common Programming Interface Communications Reference*.

A sample SCOMDIR NAMES file is shown in Figure 5, which contains a sample side information definition for the ANAME server.

```
:nick.ANAMED :luname.gateway luname
```

Figure 5. Sample Symbolic Destination Name Definition for ANAME

**:nick.**

Specifies the symbolic destination name for the ANAME server.

**:luname.**

Specifies the location of the ANAME server. The LU name consists of two fields of up to eight characters each, separated by a space. The first field contains the ANAME client's gateway. The second field contains the ANAME server's LU name.

For detailed information about the use and contents of SCOMDIR or UCOMDIR files, see *VM/ESA Connectivity Planning, Administration, and Operation*.

---

## Configuring the ANAME Server (ANAMED)

To configure the ANAME server on VM:

1. Locate the ANAME server files.
2. Define the ANAME server's virtual machine.

### Task 1 - Locate the ANAME Server Files

When the APPC Application Suite was installed, the ANAME server files should have been copied and renamed. Table 3 contains a description of each ANAME server file, the name of the file that was retrieved from the IBM Software Distribution tape (*installed file name*) and the name of the file that is used by the APPC Application Suite programs (*production file name*). Each ANAME server file must be renamed *from* the installed file name *to* the production file name.

If the ANAME server files have not been renamed, rename the files before continuing to the next step. See the *Program Directory for APPC Application Suite, VM Version 1* for additional information about copying and renaming the ANAME server files.

Table 3. ANAME Server Files

File Description	Installed File Name	Production File Name
ANAME server REXX EXEC	APPMADVX EXEC	ANAMED EXEC
ANAMED initialization file	APPMANVI INI	ANAMED INI

### Task 2 - Define the ANAME Server's Virtual Machine

The ANAME server must be installed on a dedicated virtual machine. The server's virtual machine requires:

- At least 16M of virtual storage.
- Privilege class G.

If you run the server's virtual machine as a disconnected service machine, it should be defined to your system automation so the server is started automatically. To send the ANAME halt command to the ANAME server without logging on to the server's virtual machine, set up a secondary operator. See "halt" on page 62 for information about the ANAME halt command.

To define the ANAME server's virtual machine:

- Create a virtual machine CP directory.
- Create a logon profile.

### **Create a Virtual Machine CP Directory for the ANAME Server**

Include the following statements in the ANAME server virtual machine's CP directory:

#### **IUCV \*IDENT RESANY GLOBAL**

Lets the ANAME virtual machine connect to the \*IDENT server and define itself as a system resource.

#### **IPL CMS PARM AUTO CR**

Requests automatic initialization of CMS. When the ANAME server's virtual machine is autologged, CMS is loaded without operator intervention.

#### **MACHINE ESA**

Use the MACHINE ESA statement to run the ANAME server's virtual machine in ESA mode.

#### **LINK CMSDISK**

Links the ANAME server's virtual machine to the CMS disks. The ANAME server cannot operate without CMS.

#### **MDISK 191**

Defines the number of cylinders required on the minidisk. The ANAMED files and the logon profile require approximately 10 cylinders of storage. You must also allow space for the ANAME database file. To determine how many cylinders are required for the database file, see "Estimate Storage Requirements for the ANAME Database".

For more information about CP directories, see *VM/ESA: Planning and Administration*

Figure 6 shows an example of a CP directory for the ANAME server's virtual machine.

```
USER ANAME XXXXXXXX 16M 32M G 64
ACCOUNT 49AB7A0Z
IUCV *IDENT RESANY GLOBAL
MACHINE ESA
IPL      CMS PARM AUTO CR
CONSOLE 01F 3215
SPOOL   00C 2540 READER *
SPOOL   00D 2540 PUNCH A
LINK CMSDISK 19E 19E RR ALL
LINK CMSDISK 29E 29E RR ALL
MDISK 191 3380 388 010 CMS58C MR
```

*Figure 6. Sample ANAME Virtual Machine CP Directory*

**Estimate Storage Requirements for the ANAME Database:** Before you estimate the storage requirements for the ANAME database file, you need to estimate the maximum number of records that the database file will contain. To estimate the number of database records, you need to understand how ANAME updates the database.

Before ANAME updates a record in the database, ANAME completes these steps to protect the database from damage:

1. The database file is renamed.

2. A new database file is created into which the entire database, including the change, is written.
3. If the new database file is written without errors, the old database file is deleted.

Thus, you need to allocate enough disk space to accommodate *twice* the maximum number of records you estimate the database will contain.

Each record in the ANAME database is 213 bytes. Additionally, a 4-byte header is placed at the beginning of the database file.

One 3390 cylinder holds approximately 890 000 bytes, or approximately 4178 database records. Thus, if you expect the ANAME database to contain a maximum of 10 000 records, the database file will require 5 cylinders of disk space (enough for 20 000 records). You can use this formula to determine how much disk space to allocate to the database file:

$$((\text{maximum number of records} * 213) + 4) * 2 = \text{required disk space}$$

### Create a Logon Profile

When you create the ANAME server's virtual machine, you need to create a logon profile that contains the following:

#### SET SERVER ON

Enables CMS to start the ANAMED transaction program.

**Note:** CMS cannot start the ANAME server if another program, such as FILELIST, is already active on your virtual machine.

#### ANAMED

Starts the ANAME server automatically.

**Note:** ANAMED should be the last line of the profile.

---

## Configuring the A3270 Server

To configure the A3270 server on VM:

1. Locate the A3270 server files.
2. Define the A3270 server's virtual machine.

### Task 1 - Locate the A3270 Server Files

When the APPC Application Suite was installed, the A3270 server files should have been copied and renamed. Table 4 contains a description of each A3270 server file, the name of the file that was retrieved from the IBM Software Distribution tape (*installed file name*) and the name of the file that is used by the APPC Application Suite programs (*production file name*). Each A3270 server file must be renamed *from* the installed file name *to* the production file name.

If the A3270 server files have not been renamed, rename the files before continuing to the next step. See *Program Directory for APPC Application Suite, VM Version 1*.

Table 4. A3270 Server Files

File Description	Installed File Name	Production File Name
A3270 server REXX EXEC	APP3A3VX EXEC	A3270 EXEC
A3270 server initialization file	APP3270V INI	A3270D INI



## Task 2 - Define the A3270 Server's Virtual Machine

The A3270 server must be installed on a dedicated virtual machine. The server's virtual machine requires:

- At least 16M of virtual storage.
- Privilege class G.

If you run the server's virtual machine as a disconnected service machine, it should be defined to your system automation so the server is started automatically. To send A3270 commands to the A3270 server without logging on to the server's virtual machine, set up a secondary operator. See "A3270 Server Command Reference" on page 66 for descriptions of the A3270 commands.

To define the A3270 server's virtual machine:

- Create a virtual machine CP directory.
- Create a logon profile.

### Create a Virtual Machine CP Directory for the A3270 Server

Include the following statements in the A3270 server virtual machine's CP directory:

#### IUCV \*IDENT RESANY GLOBAL

Lets the A3270 virtual machine connect to the \*IDENT server and define itself as a system resource.

#### OPTION MAXCONN 16|nnnnn

Specifies the maximum number of inter-user communication vehicle (IUCV) and APPC/VM connections that are allowed for the A3270 server. Because two IUCV connections are required for each active terminal session between an A3270 client and the A3270 server, the value assigned to *nnnnn* should be twice the maximum number of terminal sessions planned for the A3270 server. If you do not specify a maxconn value, the default is 16.

**Note:** On VM, you can set the A3270 server to accept up to a maximum of 4000 clients. The actual number of active terminal sessions allowed for the A3270 server is defined using the maxclients command in the A3270 initialization file.

If the maxclients command is set to accept 4000 clients, maxconn must be set to 8000. If maxconn is set to a value that is less than twice the value of maxclients, the maxconn value determines the maximum number of terminal sessions. For example, if maxclients is set to 4000 and maxconn is set to 6000, the A3270 server can only accept 3000 active terminal sessions.

#### IPL CMS PARM AUTOCR

Requests automatic initialization of CMS. When the A3270 virtual machine is autologged, CMS is loaded without operator intervention.

#### MACHINE ESA

Use the MACHINE ESA statement to run the A3270 virtual machine in ESA mode.

#### LINK CMSDISK

Links the A3270 server's virtual machine to the CMS disks. The A3270 server cannot operate without CMS.

#### MDISK 191

Sets aside room on the minidisk for the A3270 module, A3270 exec, and the logon profile. Approximately 10 cylinders of storage are needed.

For more information about CP directories, see *VM/ESA: Planning and Administration*

Figure 7 shows a sample CP directory for an A3270 virtual machine.

```
USER A3270 XXXXXXXX 16M 32M G 64
ACCOUNT 49AB7A0Z
IUCV *IDENT RESANY GLOBAL
OPTION MAXCONN 1024
MACHINE ESA
IPL      CMS PARM AUTOCR
CONSOLE  01F 3215
SPOOL    00C 2540 READER *
SPOOL    00D 2540 PUNCH A
LINK CMSDISK 19E 19E RR ALL
LINK CMSDISK 29E 29E RR ALL
MDISK 191 3380 388 010 CMS58C MR
```

*Figure 7. Sample A3270 VM/ESA Virtual Machine Directory*

### **Create a Logon Profile**

When you create the A3270 server's virtual machine, you need to create a logon profile that contains the following statements.

#### **SET SERVER ON**

Enables CMS to start the A3270D transaction program.

**Note:** CMS cannot start the A3270 server if another program, such as FILELIST is already active on your virtual machine.

#### **A3270**

Starts the A3270 server automatically.

**Note:** A3270 should be the last line of the profile.

---

## Chapter 3. Configuring the APPC Application Suite on MVS

This chapter describes the configuration tasks you must complete to define the APPC Application Suite to APPC/MVS. The APPC Application Suite provides four applications: AFTP, APING, ANAME, and the A3270 server. See “Product Overview” on page 1 for a brief overview of each application.

To define APPC applications to APPC/MVS, you must use the APPC/MVS Administration Utility (ATBSDFMU). Sample JCL is provided with the APPC Application Suite that, once modified, you can use to invoke the APPC/MVS Administration Utility from batch. For additional information about the APPC/MVS Administration Utility, see *z/OS MVS Planning: APPC/MVS Management*.

Because the configuration tasks for each application vary, configuration for each application is described in a separate section.

For information about configuring the APPC Application Suite applications on VM, see “Chapter 2. Configuring the APPC Application Suite on VM” on page 3.

---

### Before You Start

Before you can configure the APPC Application Suite on z/OS, you must install the APPC Application Suite messages. Additionally, ensure that:

- The required system software is installed.
- The APPC Application Suite is installed.
- The MVS Message Server (MMS) is enabled.
- APPC/MVS is started.

Instructions for installing the APPC Application Suite messages are contained in “Installing APPC Application Suite Messages”.

For information about system software requirements and APPC Application Suite installation, see the *Program Directory for APPC Application Suite - MVS Version 1*.

For information about configuring APPC/MVS, refer to *z/OS MVS Planning: APPC/MVS Management*.

### Installing APPC Application Suite Messages

The APPC Application Suite uses the MVS Messaging Service (MMS) to provide messages. Before you can use the APPC Application Suite, the APPC Application Suite messages must be installed. If MMS is not currently configured on your system, use the sample JCL provided with the APPC Application Suite to configure MMS and install the APPC Application Suite messages.

To determine whether MMS is already enabled on your system, issue the MMS command, as follows:

```
SET MMS=NO
```

One of the following messages will be displayed in response to the MMS command:

- IEE294I SET MMS COMMAND REJECTED. MVS MESSAGE SERVICE NOT ACTIVE.

If message IEE2941 is displayed, proceed to “Configuring MMS and Installing the APPC Application Suite Messages” on page 14

- CNLS003I TERMINATE SUCCESSFUL

If message CNLS003I is displayed, MMS is already configured on your system. Do not use the procedures described in this section.

## Configuring MMS and Installing the APPC Application Suite Messages

To configure MMS and install the APPC Application Suite messages:

1. Allocate the VSAM dataset for messages.
2. Compile the APPC Application Suite messages.
3. Create the MMSLSTxx PARMLIB member.
4. Modify the CONSOLxx PARMLIB member.
5. Activate MMS.

**Step 1 - Allocate the VSAM Dataset for Messages:** Allocate the MMS VSAM U.S. English messages dataset by modifying the sample JCL that is provided with the APPC Application Suite. The member name of the sample is APPZMSVJ. Follow the instructions contained in the APPZMSVJ JCL to modify it.

When the APPZMSVJ JCL is modified, submit the JCL and verify that the job completes with a return code of 0.

**Step 2 - Compile the &applsuite Messages:** Compile the APPC Application Suite messages into the MMS VSAM U.S. English message dataset by modifying the sample JCL that is provided with the APPC Application Suite. The member name of the sample is APPZMSCJ. Follow the instructions contained in the APPZMSCJ JCL to modify it.

When the APPZMSCJ JCL is modified, submit the JCL and verify that the job completes with a return code of 0.

**Step 3 - Create the MMSLSTxx PARMLIB Member:** To create the MMSLSTxx PARMLIB member, copy the sample member APPZMSPA into your system PARMLIB member and name it MMSLSTxx, where xx is a two character designation of your choice. The default name of the system PARMLIB dataset is SYS1.PARMLIB.

Thez/OS *MVS Initialization and Tuning Reference* contains information on defining the SYS1.PARMLIB CONSOLxx and MMSLSTxx members to your system to enable MMS.

**Step 4 - Modify the CONSOLxx PARMLIB Member:** To modify the CONSOLxx PARMLIB member, edit the CONSOLxx member that you use to start your system. In the INIT section, add or change the MMS() to MMS(xx), where xx is the designation that you defined in Step 3.

Once MMS(xx) is specified in your system's CONSOLxx member, MVS will automatically start MMS during the next system IPL.

**Step 5 - Activate MMS:** To activate MMS now, issue the MMS command as follows:

```
SET MMS=xx
```

Where xx is the designation that you defined in Task 3. The message CNLS003I INITIALIZE SUCCESSFUL should be displayed. This message indicates that you have successfully installed the APPC Application Suite messages, and can configure the APPC Application Suite on MVS.

---

## Configuring AFTP Clients and Servers

To configure AFTP on MVS:

1. Modify the AFTP client REXX EXEC.
2. Modify the ACOPY client REXX EXEC.
3. Modify the AFTP server REXX EXEC
4. Add side information.
5. Add a transaction program (TP) definition.

### Task 1 - Modify the AFTP Client REXX EXEC

A REXX EXEC starts the AFTP client on MVS. A sample REXX EXEC for AFTP, referred to in this section as the AFTP EXEC, is provided with the APPC Application Suite. The member name of the sample AFTP EXEC is APPFAFMX.

**Note:** Before you begin this task, determine the name of the partitioned datasets that contain:

- The APPFTP load module.
- The AFTP initialization file. The member name of the sample AFTP initialization file is APPFTPI.

To update the AFTP EXEC:

- Change *misc* to the name of the partitioned data set in which the AFTP initialization file is stored.
- Change *loadlib* to the name of the partitioned data set in which the APPFTP load module is installed.

A portion of the sample AFTP EXEC is shown in Figure 8. In the illustration, the values you need to update are *underlined>*. In the actual sample, the values *are not* underlined.

```
/* REXX for MVS to start the AFTP client. */
:

/* Allocate the initialization file. */
"ALLOC DD(APPFTPI) DS('misc(APPFTPI)') SHR REUSE"

/* Allocate the error log dataset. */
"ALLOC DD(APPFTPE) DS('"SYSVAR(SYSUID)".ASUITE.LOGS.APPFTPE')",
"MOD RECFM(V,B) LRECL(132) DSORG(PS) BLKSIZE(3120)"
if rc<>0 then do say "Dataset allocation error. RC="rc;exit;end

/* Allocate the trace dataset. */
"ALLOC DD(APPFTPT) DS('"SYSVAR(SYSUID)".ASUITE.LOGS.APPFTPT')",
"MOD RECFM(V,B) LRECL(132) DSORG(PS) BLKSIZE(3120)"
if rc<>0 then do say "Dataset allocation error. RC="rc;exit;end
:

/* Call the program */
address tso call 'loadlib(APPFTP)' parameters asis
TpRc = rc
```

Figure 8. Partial Sample of the AFTP EXEC

## Task 2 - Modify the ACOPIY Client REXX EXEC

A REXX EXEC starts the ACOPIY client on MVS. A sample REXX EXEC for ACOPIY, referred to in this section as the ACOPIY EXEC, is provided with the APPC Application Suite. The member name of the sample ACOPIY EXEC is APPFACMX.

**Note:** Before you begin this task, determine the name of the partitioned datasets that contain:

- The APPCOPY load module.
- The AFTP initialization file. The member name of the sample AFTP initialization file is APPFTPI.

To update the ACOPIY EXEC:

- Change *misc* to the name of the partitioned data set in which the AFTP initialization file is stored.
- Change *loadlib* to the name of the partitioned data set in which the APPCOPY load module is installed.

A portion of the sample ACOPIY EXEC is shown in Figure 9. In the illustration, the values you need to update are *underlined>*. In the actual sample, the values are *not* underlined.

```
⋮  
  
/* Allocate the initialization file. */  
"ALLOC DD(APPFTPI) DS('misc(APPFTPI)') SHR REUSE"  
  
/* Allocate the error log dataset. */  
"ALLOC DD(APPCOPYE) DS('SYSVAR(SYSUID).ASUITE.LOGS.APPCOPYE')",  
    "MOD RECFM(V,B) LRECL(132) DSORG(PS) BLKSIZE(3120)"  
if rc<>0 then do say "Dataset allocation error. RC="rc;exit;end  
  
/* Allocate the trace dataset. */  
"ALLOC DD(APPCOPYT) DS('SYSVAR(SYSUID).ASUITE.LOGS.APPCOPYT')",  
    "MOD RECFM(V,B) LRECL(132) DSORG(PS) BLKSIZE(3120)"  
if rc<>0 then do say "Dataset allocation error. RC="rc;exit;end  
⋮  
  
/* Call the program */  
address tso call 'loadlib(APPCOPY)'' parameters asis  
TpRc = rc
```

Figure 9. Partial Sample of the ACOPIY EXEC

## Task 3 - Modify the AFTP Server REXX EXEC

A REXX EXEC starts the AFTP server (AFTPD) on MVS. A sample REXX EXEC for AFTPD, referred to in this section as the AFTPD EXEC, is provided with the APPC Application Suite. The member name of the sample AFTPD EXEC is APPFADMX.

**Note:** Before you begin this task, determine the name of the partitioned datasets that contain:

- The AFTPD EXEC.
- The APPFTP load module.

To update the AFTPD EXEC:

- Change *misc* to the name of the partitioned dataset in which the APPFADMX member is stored.

- Change *loadlib* to the name of the partitioned data set in which the APPFTP load module is installed.

A portion of the sample AFTPD EXEC is shown in Figure 10. In the illustration, the values you need to update are *underlined>*. In the actual sample, the values *are not* underlined.

```

/* REXX FOR MVS TO START THE AFTP SERVER. */
:

/* ALLOCATE THE INITIALIZATION FILE. */
"ALLOC DD(APPFTPI) DS('PRF.ASUITE.misc(APPFTPI)') SHR REUSE"
IF RC<>0 THEN DO SAY "APPFTPDE ALLOCATION ERROR. RC="RC;EXIT;END
:

/* CALL THE PROGRAM */
ADDRESS TSO CALL "'loadlib(APPFTPD)'" ASIS
TPRC = RC
:

EXIT TPRC

```

Figure 10. Partial Sample of the AFTPD EXEC

**Notes:**

1. When an AFTP or ACOPY client specifies a user ID of anonymous, the AFTPD trace, error and audit logs are allocated to SYSOUT. If you want these logs to go to datasets, customize the three allocate statements that follow this statement in the APPFADM sample JCL:  

```
/* CASE 2: USER IS ANONYMOUS - ALLOCATE ALL DATASETS TO SYSOUT */
```
2. If you want to view the error log and trace log datasets while the transactions are running, you can preallocate the datasets and change the DISP parameter to DISP=(SHR,KEEP,KEEP).
3. Allocate larger trace log datasets if support personnel indicate that you will need to trace many transactions, or if you are asked to set the trace level to 100 or higher.
4. If you will never use the trace facility, you can remove the trace log dataset allocation.
5. Error logs should occasionally be purged of outdated information.
6. Increase the size of the audit log on servers that will experience a high transaction rate.
7. Audit logs should occasionally be purged of outdated information.
8. You can keep one audit log per system instead of one audit log per server by changing the sample JCL as follows:
  - a. Allocate an audit log dataset large enough to accommodate audit log entries for every AFTP server installed on the system.
  - b. Authorize all system users for write access to the audit log dataset.
  - c. Allocate APPFTPDA to the new audit log dataset.

- If you do not want to keep the AFTP audit log, allocate APPFTPDA to SYSOUT.

## Task 4 - Add Side Information

Side information is required when AFTP is installed on releases of MVS prior to MVS Version 4 Release 3. You do not need AFTP side information when AFTP is installed on MVS Version 4 Release 3 or later versions.

Side information can be defined to APPC/MVS using JCL that contains the SIADD utility command and its associated keywords. Sample side information JCL is provided for AFTP. The member name of the sample is APPFSIAJ. It is illustrated in Figure 11. In the illustration, the value that you need to modify is *underlined*. In the actual sample, this value is *not* underlined.

**Note:** Before you begin this task, determine the name of the partitioned data set that contains the VSAM side information.

To update the APPFSIAJ JCL, change *appcsi* to the name of the VSAM side information dataset.

The SIADD command keywords shown in the example are described below.

```
//APPFSIAJ JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID;
//*****
//*
//* THIS EXAMPLE ADDS SIDE INFORMATION FOR AFTP
//*
//*
//* To customize for your system:
//*
//* 1) Change "appcsi" to the name of your APPC/MVS side
//*    information dataset. The APPC/MVS side information
//*    dataset is often named SYS1.APPCSI or USER.APPCSI.
//*
//* 2) Change "SIADD" to SIMODIFY if this definition
//*    already exists and you wish to change it.
//*
//*****
//STEP      EXEC PGM=ATBPDFMU
//SYSPRINT DD  SYSOUT=*
//SYSSDLIB DD  DSN=appcsi,DISP=SHR
//SYSSDOUT DD  SYSOUT=*
//SYSIN      DD  *
              SIADD
                  DESTNAME(AFTP)
                  TPNAME(AFTP)
```

Figure 11. Sample AFTP Side Information JCL

### DESTNAME(AFTP)

Specifies the AFTP server's symbolic destination name.

### TPNAME(AFTP)

Specifies the AFTP server's transaction program name.

**Note:** Although TPNAME is a required keyword, the keyword value is ignored by the AFTP clients.

See *z/OS MVS Planning: APPC/MVS Management* for detailed information about side information.



## Modifying AFTP Side Information

If you need to change the AFTP side information at a later time, change the SIADD command in the side information JCL to SIMODIFY.

## Task 5 - Add a Transaction Program Profile for the AFTP Server

A transaction program (TP) profile must be added for the AFTP server (AFTPD). The TP profile defines AFTPD to APPC/MVS as a standard transaction program, and enables the APPC/MVS transaction scheduler to schedule the AFTPD transaction program when an inbound allocation request is received.

Transaction program profiles can be defined to APPC/MVS using JCL that contains the TPADD utility command and its associated keywords. Sample TP profile JCL is provided for AFTP. The member name of the sample is APPFTPAJ. A portion of the sample is illustrated in Figure 12 on page 20. In the illustration, the values that you need to modify are *underlined*. In the actual sample, these values *are not* underlined.

**Note:** Before you begin this task, determine the names of the partitioned data sets that contain:

- The VSAM TP profile.
- The APPFADMJ member. The APPFADMJ member is the REXX EXEC that starts the AFTP server.

To update the APPFTPAJ JCL:

- Change *appctp* to the name of your VSAM TP profile dataset.
- Change *clist* to the name of the partitioned dataset in which the APPFADMJ member is installed.

```

:

//STEP      EXEC PGM=ATBSDFMU
//SYSPRINT DD  SYSOUT=*
//SYSSDLIB DD  DSN=appctp,DISP=SHR
//SYSSDOUT DD  SYSOUT=*
//SYSIN      DD  DATA,DLM=XX
      TPADD
          TPNAME(AFTPD)
          ACTIVE(YES)
          TPSCHED_DELIMITER(##)
          TAILOR_SYSOUT(NO)
          TAILOR_ACCOUNT(NO)
          CLASS(FAST)
          TPSCHED_TYPE(STANDARD)
          KEEP_MESSAGE_LOG(ERROR)
          MESSAGE_DATA_SET(&SYSUID;AFTPD.JOBLOG)
          DATASET_STATUS(MOD)
          JCL_DELIMITER(END_OF_JCL)
//AFTPD     JOB  MSGLEVEL=(1,1),MSGCLASS=A
//RUNAFTPD EXEC PGM=IKJEFT01,DYNAMNBR=50,REGION=6M,
          PARM='EXEC 'c l i s t (APPFADMX) '

:

END_OF_JCL
##
XX
/*

```

Figure 12. Partial Sample AFTPD TP Add JCL

**Note:** Ensure that CLASS(FAST) is defined to the APPC/MVS scheduler.

For additional information about the TPADD command and its associated keywords, or about the transaction scheduler, see *z/OS MVS Planning: APPC/MVS Management*

### Modifying the AFTPD TP Profile

If you need to change the AFTPD TP profile at a later time, change the TPADD command in the APPFTPAJ sample

---

## Configuring APING Clients and Servers

To configure &aping on MVS:

1. Modify the APING client REXX EXEC
2. Add side information
3. Add a transaction program (TP) definition

### Task 1 - Modify the APING Client REXX EXEC

A REXX EXEC starts the APING client on MVS. A sample REXX EXEC for APING, referred to in this section as the APING EXEC, is provided with the APPC Application Suite. The member name of the sample APING EXEC is APPPAPMX. It is illustrated in Figure 13 on page 21. In the illustration, the value you need to update is *underlined*. In the actual sample, the value *is not* underlined.

**Note:** Before you begin this task, determine the name of the partitioned dataset that contains the APPPING load module.

To update the APING EXEC, change *loadlib* to the name of the partitioned data set in which the APPPING loadlib is installed.

```

/* REXX for MVS to start the APING client. */

/*****
 *
 * To customize for your environment, change "loadlib"
 * in the "address tso call..." line to the location where
 * the APPPING loadlib is installed.
 *
 *****/

/* Allocate the APING error log. */
"ALLOC DD(APPPINGE) DS('SYSVAR(SYSUID).ASUITE.LOGS.APPPINGE')",
"MOD RECFM(V,B) LRECL(132) DSORG(PS) BLKSIZE(3120)"
if rc<>0 then do say "Dataset allocation error. RC="rc;exit;end

/* Allocate the APING trace log. */
"ALLOC DD(APPPINGT) DS('SYSVAR(SYSUID).ASUITE.LOGS.APPPINGT')",
"MOD RECFM(V,B) LRECL(132) DSORG(PS) BLKSIZE(3120)"
if rc<>0 then do say "Dataset allocation error. RC="rc;exit;end

arg parameters
if parameters <> '' then parameters = ""parameters""

address tso call 'loadlib(APPPING)'' parameters
TpRc = rc
:

exit TpRc

```

Figure 13. Sample APING EXEC

## Task 2 - Add Side Information

Side information is required when APING is installed on releases of MVS prior to MVS Version 4 Release 3. You do not need APING side information when APING is installed on MVS Version 4 Release 3 or later.

Side information can be defined to APPC/MVS using JCL that contains the SIADD utility command and its associated keywords. Sample side information JCL is provided for APING. The member name of the sample is APPPSIAJ. It is illustrated in Figure 14 on page 22. In the illustration, the value that you need to modify is *underlined*. In the actual sample, this value *is not* underlined.

**Note:** Before you begin this task, determine the name of the partitioned data set that contains the VSAM side information.

To update the APPPSIAJ JCL, change *appcsi* to the name of the VSAM side information dataset.

```

//APPSIAJ JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID;
//*****
/*
/* THIS EXAMPLE ADDS SIDE INFORMATION FOR THE APING CLIENT
/*
/* If you have MVS V4R3, then you do not need to add side
/* information. You must add side information in MVS releases
/* prior to V4R3 because they do not support a feature known
/* as a "CPI-C blank symbolic destination name".
/*
/* To customize for your system:
/*
/* 1) Change "appcsi" to the name of your APPC/MVS side
/* information dataset. The APPC/MVS side information
/* dataset is often named SYS1.APPCSI or USER.APPCSI.
/*
/* 2) Change "SIADD" to SIMODIFY if this definition
/* already exists and you wish to change it.
/*
//*****
//STEP EXEC PGM=ATBDSDFMU
//SYSPRINT DD SYSOUT=*
//SYSSDLIB DD DSN=appcsi,DISP=SHR
//SYSSDOUT DD SYSOUT=*
//SYSIN DD *
        SIADD
                DESTNAME(APINGD)
                TPNAME(APINGD)
/*

```

Figure 14. Sample APING Add Side Information JCL

The SIADD command keywords shown in the example are described below.

**DESTNAME(APINGD)**

Specifies the APING server's symbolic destination name.

**TPNAME(APINGD)**

Specifies the APING server's transaction program name.

**Note:** Although TPNAME is a required keyword, the keyword value is ignored by the APING client.

See *z/OS MVS Planning: APPC/MVS Management* for detailed information about side information.

**Modifying APING Side Information**

If you need to change the APING side information at a later time, change the SIADD command in the side information JCL to SIMODIFY.

**Task 3 - Add a Transaction Program (TP) Profile**

A TP profile for APINGD can be defined to APPC/MVS using JCL that contains the TPADD utility command and its associated keywords. Sample TP profile JCL is provided for APINGD. The member name of the sample TP profile is APPPTPAJ. A portion of the sample is illustrated in Figure 15 on page 23. In the illustration, the values that you need to modify are *underlined*. In the actual sample, these values are *not* underlined.

**Note:** Before you begin this task, determine the names of the partitioned data sets that contain:

- The VSAM TP profile.

- The APPPING load module.
- The C/370\* runtime libraries. The member names of the C/370 runtime libraries are SEDCLINK and SIMBLINK.

To update the APPPTPAJ JCL:

- Change *appctp* to the name of your VSAM TP profile dataset.
- Change *loadlib* to the name of the partitioned dataset in which the APPC Application Suite load modules are installed.
- Change *sedclink* to the name of the partitioned dataset in which the C/370 runtime SEDCLINK member is installed.
- Change *sibmlink* to the name of the partitioned dataset in which the C/370 runtime SIBMLINK member is installed.
- Change *logs* to the name of the partitioned dataset in which the APINGD error log should be stored.

⋮

```
//STEP      EXEC PGM=ATBSDFMU
//SYSPRINT DD  SYSOUT=*
//SYSSDLIB DD  DSN=appctp,DISP=SHR
//SYSSDOUT DD  SYSOUT=*
//SYSIN     DD  DATA,DLM=XX
           TPADD
           TPNAME(APINGD)
           ACTIVE(YES)
           TPSCHED_DELIMITER(##)
           TAILOR_SYSOUT(NO)
           TAILOR_ACCOUNT(NO)
           CLASS(FAST)
           TPSCHED_TYPE(STANDARD)
           KEEP_MESSAGE_LOG(ERROR)
           MESSAGE_DATA_SET(&SYSUID;APINGD.JOBLOG)
           DATASET_STATUS(MOD)
           JCL_DELIMITER(END_OF_JCL)
//APINGD   JOB  MSGLEVEL=(1,1),MSGCLASS=0
//RUNAPNGD EXEC PGM=APPPINGD
//STEPLIB DD DSN=loadlib,DISP=SHR
//         DD DSN=sedclink,DISP=SHR
//         DD DSN=sibmlink,DISP=SHR
//*
//* Allocate the error log file.
//APPPNGDE DD DSN=logs.APPPNGDE,
//         DISP=(MOD,CATLG,CATLG),UNIT=SYSALLDA,SPACE=(CYL,2)
//*
//SYSPRINT DD SYSOUT=A,FREE=CLOSE
END_OF_JCL
##
XX
/*
```

Figure 15. Partial Sample APINGD Add TP Profile JCL

#### Notes:

1. Ensure that CLASS(FAST) is defined to the APPC/MVS scheduler.
2. If you want to view the error log and trace log datasets while the transactions are running, you can preallocate the datasets and change the DISP parameter to DISP=(SHR,KEEP,KEEP).

3. Error logs should occasionally be purged of outdated information.

For additional information about the TPADD command and its associated keywords, or about the transaction scheduler, see *z/OS MVS Planning: APPC/MVS Management*

### **Modifying the APINGD TP Profile**

If you need to change the APINGD TP profile at a later time, change the TPADD command in the APPPTPAJ sample JCL to TPMODIFY.

---

## **Configuring ANAME Clients**

To configure the ANAME client on MVS:

1. Modify the ANAME client REXX EXEC
2. Add side information.

### **Task 1 - Modify the ANAME Client REXX EXEC**

A REXX EXEC starts ANAME on MVS. A sample REXX EXEC for ANAME, referred to in this section as the ANAME EXEC, is provided with the APPC Application Suite. The member name of the sample REXX EXEC for ANAME is APPMANMX. It is illustrated in Figure 16 on page 25. In the illustration, the value you need to change is *underlined*. In the actual sample, this value is *not* underlined.

**Note:** Before you begin this task, determine the name of the partitioned dataset in which APPNAME load module is installed.

To update the ANAME EXEC, change *loadlib* to the name of the partitioned data set in which the APPNAME load module is installed.

```

/* REXX for MVS to start the ANAME client. */

/*****
 *
 * To customize for your environment, change "loadlib"
 * in the "address tso call..." line to the location where
 * the APPNAME loadlib is installed.
 *
 *****/

/* Allocate the error log dataset. */
"ALLOC DD(APPNAMEEE) DS('SYSVAR(SYSUID).ASUITE.LOGS.APPNAMEE')",
"MOD RECFM(V,B) LRECL(132) DSORG(PS) BLKSIZE(3120)"
if rc<>0 then do say "Dataset allocation error. RC="rc;exit;end

/* Allocate the trace dataset. */
"ALLOC DD(APPNAMEET) DS('SYSVAR(SYSUID).ASUITE.LOGS.APPNAMEET')",
"MOD RECFM(V,B) LRECL(132) DSORG(PS) BLKSIZE(3120)"
if rc<>0 then do say "Dataset allocation error. RC="rc;exit;end
:
:

/* Call the program */
address tso call 'loadlib(APPNAME)'' parameters asis
TpRc = rc
:
:

exit TpRc

```

Figure 16. Sample ANAME EXEC

## Task 2 - Add Side Information

The symbolic destination name for your ANAME server must be added to APPC/MVS side information. ANAME side information can be defined to APPC/MVS using JCL that contains the SIADD utility command and its associated keywords. Sample side information JCL is provided for APING. The member name of the sample is APPMSIAJ. It is illustrated in Figure 17 on page 26. In the illustration, the values that you need to modify are *underlined*. In the actual sample, these values are *not* underlined.

**Note:** Before you begin this task, determine:

- The name of the partitioned data set that contains the VSAM side information.
- The fully qualified LU name of the computer on which the ANAME server is installed.

To update the APPMSIAJ JCL:

- Change *appcsi* to the name of the VSAM side information dataset.
- Change *netid.luname* to the fully qualified LU name of the computer on which the ANAME server is installed.

```

/*-----*/
//APPMIAJ JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID;
//*****
:

//STEP EXEC PGM=ATBSDFMU
//SYSPRINT DD SYSOUT=*
//SYSSDLIB DD DSN=appcsi,DISP=SHR
//SYSSDOUT DD SYSOUT=*
//SYSIN DD *
SIADD
DESTNAME(ANAMED)
TPNAME(ANAMED)
PARTNER_LU(netid.luname)

```

Figure 17. Sample ANAME Side Information JCL

The SIADD command keywords shown in the example are described below.

**DESTNAME(ANAMED)**

The ANAME server's symbolic destination name is ANAMED.

**TPNAME(ANAMED)**

The ANAME server's transaction program name is ANAMED.

**Note:** Although TPNAME is a required keyword, the keyword value is ignored by the ANAME client.

**PARTNER\_LU(netid.luname)**

The fully qualified LU name of the computer on which the ANAME server is installed.

See *z/OS MVS Planning: APPC/MVS Management* for detailed information about side information.

**Modifying ANAME Side Information**

If you need to change the ANAME side information at a later time, change the SIADD command in the side information JCL to SIMODIFY.

## Configuring the ANAME Server

To configure the ANAME server on MVS, you need to add a transaction program (TP) profile.

### Task 1 - Add a Transaction Program (TP) Profile

A transaction program (TP) profile must be added for the ANAME server. The TP profile defines ANAMED to APPC/MVS as a system-level, multi-trans transaction program, and enables the APPC/MVS transaction scheduler to schedule the ANAMED transaction program when an inbound allocation request is received.

A TP profile for ANAMED can be added to the APPC/MVS TP profile file using JCL that contains the TPADD utility command and its associated keywords. Sample TP profile JCL is provided for ANAME, which is named APPMTPAJ. A portion of the sample is illustrated in Figure 18 on page 28. In the illustration, the values that you need to modify are *underlined*. In the actual sample, these values *are not* underlined.



**Note:** Before you begin this task, to determine the locations of the datasets that contain:

- The VSAM TP profile.
- The ANAMED initialization file. The member name of the sample ANAME initialization file is APPMANDI.
- The APPNAMED load module.
- The C/370 runtime libraries. The C/370 runtime libraries are named SEDCLINK and SIBMLINK.
- The ANAME database. The dataset name of the ANAME database is APPNAMDF.

To update the APPMTPAJ JCL:

- Change *appctp* to the name of your VSAM TP profile dataset.
- On the GENERIC\_ID parameter, change *user\_id* to a special user ID that can be used for multi-trans shell processing. See “Notes” on page 28 for more information.
- Change *loadlib* to the name of the partitioned dataset in which the APPNAMED load module is installed.
- Change *sedclink* to the name of the partitioned dataset in which the C/370 runtime SEDCLINK library is installed.
- Change *sibmlink* to the name of the partitioned dataset in which the C/370 runtime SIBMLINK library is installed.
- Change *misc* to the name of the partitioned dataset in which the ANAMED initialization file is installed.
- Change *logs* to the name of the partitioned dataset in which your ANAMED error log should be stored.
- Change *logs* to the name of the partitioned dataset in which your ANAMED trace log should be stored.
- Change *database* to the name of the sequential dataset in which the ANAME database is stored.

```

:
//STEP      EXEC PGM=ATBSDFMU
//SYSPRINT DD  SYSOUT=*
//SYSSDLIB DD  DSN=appctp,DISP=SHR
//SYSSDOUT DD  SYSOUT=*
//SYSIN      DD  DATA,DLM=XX
          TPADD
            TPNAME(ANAMED)
            ACTIVE(YES)
            TPSCHED_DELIMITER(##)
              GENERIC_ID(userid)
              TAILOR_SYSOUT(NO)
              TAILOR_ACCOUNT(NO)
              CLASS(ANAMED)
              TPSCHED_TYPE(MULTI_TRANS)
              KEEP_MESSAGE_LOG(ERROR)
              MESSAGE_DATA_SET(&SYSUID;ANAMED.JOBLOG)
              DATASET_STATUS(MOD)
              JCL_DELIMITER(END_OF_JCL)
//ANAMED    JOB  MSGLEVEL=(1,1),MSGCLASS=0
//RUNANAMD EXEC PGM=APPNAMED
//STEPLIB  DD  DSN=loadlib,DISP=SHR
//          DD  DSN=sedclink,DISP=SHR
//          DD  DSN=sibmlink,DISP=SHR
//          /*
//          /* Allocate the initialization file.
//APPNAMDI DD  DSN=misc(APPNAMDI),DISP=SHR
//          /*
//          /* Allocate the error log file.
//APPNAMDE DD  DSN=logs.APPNAMDE,DISP=(MOD,CATLG,CATLG),
//          UNIT=SYSALLDA,SPACE=(CYL,2)
//          /*
//          /* Allocate the trace file.
//APPNAMDT DD  DSN=logs.APPNAMDT,DISP=(MOD,CATLG,CATLG),
//          UNIT=SYSALLDA,SPACE=(CYL,2)
//          /*
//          /* Allocate the ANAMED database.
//APPNAMDF DD  DSN=database,DISP=SHR
//          RECFM=F,LRECL=213,BLKSIZE=213
//SYSPRINT DD  SYSOUT=*,FREE=CLOSE
END_OF_JCL
##
XX
/*

```

Figure 18. Partial Sample ANAME Add TP Profile JCL

#### Notes:

1. The class assigned to the ANAMED program in APPMTPAJ is ANAMED. It is recommended that the ANAME server run in a class that is used only by the server. No other transaction programs should be configured to use the ANAMED class. To use CLASS(ANAMED), you must update the parmliib member ASCHPMxx using a CLASSADD statement, as follows:

```

CLASSADD
  CLASSNAME(ANAMED)
  MAX(1)
  MIN(1)
  RESPGOAL(.1)

```

See *z/OS MVS Planning: APPC/MVS Management* for detailed information about classes.

2. The GENERIC\_ID parameter defines a special user ID that can be used for multi-trans shell processing. Because the generic ID handles processing that must be isolated from different conversation partners, the generic ID must be secure from unauthorized specification or modification. To protect the generic ID, use RACF\* or an equivalent security subsystem to control read and update access to the ANAME TP profile.  
See *z/OS MVS Planning: APPC/MVS Management* for detailed information about generic user IDs.
3. If you want to view the error log and trace log datasets while the transactions are running, you can preallocate the datasets and change the DISP parameter to DISP=(SHR,KEEP,KEEP).
4. Allocate larger trace log datasets if support personnel indicate that you will need to trace many transactions, or if you are asked to set the trace level to 100 or higher.
5. If you will never use the trace facility, you can remove the trace log dataset allocation.
6. Error logs should occasionally be purged of outdated information.
7. The ANAME database should be backed up regularly. See “Backing Up the ANAME Database” on page 55 for information about backing up the ANAME database.

For additional information about the TPADD command and its associated keywords, or about the transaction scheduler, see *z/OS MVS Planning: APPC/MVS Management*

### Modifying the ANAMED TP Profile

If you need to change the ANAMED TP profile at a later time, change the TPADD command in the APPMTPAJ sample JCL to TPMODIFY.

---

## Configuring the A3270 Server

To configure the A3270 server software on MVS:

1. Define the A3270 server to VTAM.
2. Configure VTAM to start the A3270 server automatically.
3. Activate the A3270 application program major node.
4. Modify the A3270 server MVS PROC

### Task 1 - Define the A3270 Server to VTAM

The A3270 server is a VTAM application that is defined to VTAM within a VTAM application program major node definition. A sample A3270 APPL is provided with the APPC Application Suite. The member name of the sample is APP3APPL. The sample is illustrated in Figure 19 on page 32. In the illustration, the values that you need to modify are *underlined*. In the actual sample, these values *are not* underlined.

**Note:** VTAM must activate the A3270 APPL before the A3270 can successfully initialize.

The A3270 APPL defines:

- The application access method control block (ACB) name for the A3270 server. The A3270 server’s ACB name is also used as the server’s APPC LU name.
- LU type 2 (LU 2) logical terminal ACBs.

To define these functions, the A3270 VTAM APPL definition must contain the following statements:

#### **VBUILD Definition Statement**

The VBUILD definition statement defines the A3270 server to VTAM as an application program major node.

The required parameters for this statement are:

#### **TYPE=APPL**

TYPE must be set to APPL; this indicates that this VBUILD statement defines an application program major node.

#### **APPL Definition Statement for APPC**

The APPL statement for APPC defines the A3270 server's ACBNAME. The A3270 server's default ACBNAME is A3270D. If you run more than one A3270 server, define a unique ACBNAME for each server.

**Note:** The ACBNAME you define on this statement is the APPC logical unit (LU) name that A3270 clients will use to allocate APPC conversations to the A3270 server. The server's LU name must be unique within an SNA network.

The required operands for this statement are:

#### **APPC=YES**

APPC must be set to YES; allows the A3270 server to use APPC/VTAM functions.

#### **SRBEXIT=NO**

SRBEXIT must be set to NO; the A3270 server does not support SRBEXIT.

#### **DSESLIM=*max\_sessions***

Specifies the maximum number of APPC sessions that are allowed between the A3270 server and the A3270 client LUs. Because two APPC connections are required for each active terminal session between the A3270 client LUs and the A3270 server, the value assigned to *max\_sessions* should be twice the number of required terminal sessions.

Any even decimal integer in the range of 0 to 32766 is allowed. The default for the A3270 server is 50.

#### **APPL Definition Statement for LU type 2**

You need one LU type 2 (LU 2) definition for each client terminal session that the A3270 server can accept. The LU 2 ACBNAMEs you define on these statements are used by the A3270 server to establish LU 2 sessions with host applications on behalf of A3270 clients. Each LU 2 name defined on an APPL statement must be unique within a SNA network.

#### **Notes:**

1. The number of LU 2s you define should equal the number of sessions defined in the MAXCLIENTS statement in the A3270 initialization file.
2. Each LU 2 name defined in the A3270 server's application program major node definition must also be listed in the A3270 initialization file. Thus, the LU 2 names defined in the A3270 server's major node definition and the LU 2 names listed in the A3270 initialization file must match.

The required operands for this statement are:

**AUTH=NVPACE**

AUTH must be set to NVPACE.

**EAS=1**

EAS should be set to 1 to optimize VTAM resource utilization.

**PARSESS=NO**

PARSESS must be set to NO; restricts the A3270 server to one LU 2 session between itself and a host application for a given LU2 application minor node.

**MODETAB=*mode\_table\_name***

Specifies the name of the mode table to use for the A3270 server.

**Note:** It is recommended that you use the default VTAM logon mode table. The default VTAM logon mode table is named ISTINCLM.

**SESSLIM=YES**

SESSLIM must be set to YES.

The following sample VTAM application program major node definition illustrates the definition statements and operands that are required for the A3270 server. For additional information about defining VTAM application program major nodes, see the *z/OS Communications Server: SNA Resource Definition Reference*.

```

*****
*   THIS IS THE APPL MAJOR NODE FOR THE APPC 3270 MVS
*   SERVER.
*
*****
          VBUILD TYPE=APPL
*
*
* THE FOLLOWING DEFINITION IS FOR THE A3270 APPC LU
* TO WHICH ALL CLIENTS WILL ALLOCATE CONVERSATIONS.
* THE NAME SPECIFIED HERE MUST MATCH THE ACBNAME
* SPECIFIED IN THE A3270 INITIALIZATION FILE.
*
*
A3270D      APPL APPC=YES,SRBEXIT=NO,DSESLIM=50
*
*
* THE FOLLOWING DEFINITIONS ARE FOR THE LU2 LOGICAL
* TERMINALS. ONE OF THESE WILL BE USED FOR EACH
* LU2 SESSION TO AN APPLICATION ON BEHALF OF AN
* A3270 CLIENT.
*
* PLEASE NOTE THAT THE APPL DEFINITIONS BELOW
* MUST MATCH THE LU DEFINITIONS IN THE DEFAULTLUS
* STANZA OF THE A3270 INITIALIZATION FILE. IF YOU
* DEFINE LUS IN THE A3270 INITIALIZATION FILE WHICH
* ARE NOT DEFINED HERE, THEN A3270 CLIENTS WILL FAIL.
*
*
A32T01      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES
A32T02      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES
A32T03      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES
A32T04      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES
A32T05      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES
A32T06      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES
A32T07      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES
A32T08      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES
A32T09      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES
A32T10      APPL AUTH=NVPACE,EAS=1,PARSESS=NO,MODETAB=ISTINCLM,      X
              SESSLIM=YES

```

Figure 19. Sample A3270 VTAM Application Program Major Node Definition

## Using the APP3APPE Utility to Create the A3270 Application Program Major Node

The APP3APPE utility, which is supplied with the APPC Application Suite, is a TSO/REXX EXEC that creates a VTAM application major node for the MVS A3270 server. By specifying a few parameters, you can create an A3270 APPL from your TSO ID which contains:

- An APPL statement for APPC that defines the server's ACBNAME.
- APPL statements for LU 2s that define ACBNAMEs for the LU 2s.

The member name of the A3270 APPL definition that APP3APPE creates is A3270APPL.

To use the APP3APPE utility, type:

```
EXEC 'name(APP3APPE)' 'appc_acb lu2_acb_prefix num_lu2s'
```

where:

- name is the name of the partitioned dataset (PDS) in which the APP3APPE EXEC is installed.
- appc\_acb is the A3270 server's ACB name.
- lu2\_prefix is LU 2 ACB name prefix.
- num\_lu2s is the number of LU 2s you want to create.

The A3270APP utility creates the APPL in a PDS named *your\_id.vtamlst(A3270APPL)*, where *your\_id* is your TSO user ID. Copy the A3270 APPL from *your\_id.vtamlst* to your production vtamlst partitioned dataset.

For example, if you enter:

```
EXEC 'YOURID(A3270APP)' 'A3270D A32T 10'
```

A3270APP creates a VTAM application major node with a member name of A3270APPL which contains:

- An APPC APPL statement containing the A3270 server's ACB name, which is A3270D. A3270D is also the server's APPC LU name.
- 10 APPL statements that define 10 LU 2s. The ACBNAME of each LU 2 begins with the prefix **A32T** followed by a number ranging from 1 to 10. The ACBNAMEs are numbered consecutively.

## Task 2 - Configure VTAM to Start the A3270 Server Automatically

Add the A3270 server's application program major node name to the ATCCONxx member so VTAM will activate the major node during initialization.

For additional information about the ATCCONxx member, see the *z/OS Communications Server: SNA Network Implementation Guide*

## Task 3 - Start the A3270 Application Program Major Node

Before continuing to the next task, start the A3270 APPL using the VTAM activate command. See *VTAM Operation* for information about the VTAM activate command.

## Task 4 - Modify the A3270 Server MVS PROC

A PROC starts the A3270 server on MVS. A sample MVS PROC is provided with the APPC Application Suite. The member name of the sample (APP3A3MP) should have been changed to A3270D during installation. If the sample MVS PROC was not renamed, rename it now.

**Note:** Before you begin this task, determine the names of the partitioned datasets that contain:

- The APPC Application Suite load modules.
- The C/370 SEDCLINK library.
- The PLI SIMBLINK library.
- The A3270 initialization file. The member name of the sample A3270 initialization file is APP3270I.

Before you use the sample MVS PROC to start the A3270 server, change the following values:

- Change *asuite\_loadlib* to the name of the partitioned dataset that contains the APPC Application Suite load modules.
- Change *sedclink* to the name of the partitioned dataset that contains the C/370 SEDCLINK library.
- Change *sibmlink* to the name of the partitioned dataset that contains the PLI SIBMLINK library.
- Change *a3270ini* to the name of the partitioned dataset that contains the A3270 initialization file.

The sample MVS PROC is shown in Figure 20. In the illustration, the values you need to modify are *underlined*. In the actual sample, these values are *not* underlined.

```
//APP3270D PROC
//*
//* This PROC starts the A3270D server.
//*
//* Customize the following items:
//*
//*   asuite_loadlib - The dataset where the A3270 LOADLIB resides.
//*
//*   sedclink - The C/370 SEDCLINK dataset.
//*
//*   sibmlink - The PLI SIBMLINK dataset which comes with C/370.
//*
//*   a3270ini - The dataset containing the A3270 server
//*             initialization file (APP3270I).
//*
//A3270D   EXEC PGM=APP3270D,TIME=1440
//STEPLIB DD DSN=asuite_loadlib,DISP=SHR
//         DD DSN=sedclink,DISP=SHR
//         DD DSN=sibmlink,DISP=SHR
//APP3270I DD DSN=a3270ini(APP3270I),DISP=SHR
//SYSUDUMP DD SYSOUT=*
```

Figure 20. Sample A3270 MVS PROC

**Note:** You should define the A3270 MVS PROC to your system automation so that the server starts automatically after the A3270 application program major node is activated.



---

## Chapter 4. Setting Up Initialization Files

Initialization files allow you define default settings for the AFTP programs (AFTP, AFTPD, and ACOPY), the ANAME server, and the A3270 server. This section describes how to set up the initialization files for the APPC Application Suite programs.

**Note:** APING does not use an initialization file.

---

### The AFTP Initialization File

Using the AFTP initialization file allows you to:

- Define a consistent set of file transfer defaults.
- Start the trace facility automatically for all the AFTP programs.

The AFTP programs do not require the initialization file, but if the file is not found, AFTP will use default settings, as follows:

- File transfer defaults are not defined.
- The trace facility is not started automatically when an AFTP program starts.

AFTP initialization files can be placed in a common location for all users to access, or users can place a customized initialization file on their personal disk or dataset.

A sample AFTP initialization file is provided with the APPC Application Suite.

- On VM, when the sample file was installed, it was named APPFAFVI INI. The sample should have been renamed to AFTP INI.
- On MVS, when the sample file was installed, it was named APPFAFVI. The sample should have been renamed to APPFTPI.

If the sample initialization file for your operating system has not been renamed, rename it now.

The sample ANAME initialization file is illustrated in Figure 21 on page 37.

### Syntax

In the AFTP initialization file, *verbs* are followed by parameter values. Parameter values are enclosed within parentheses. A semi-colon (;) is used after the last parameter value to indicate the end of the verb. Verbs and parameters are *not* case sensitive. The correct verb syntax is:

```
verb_name
  parameter1(value1)
  parameter2(value2)
  .
  .
  parameterN(valueN);
```

Blanks, carriage returns, and tabs are ignored. You can use comments in initialization the files (*/\* \*/*), but comments cannot be nested (*/\* /\* \*/*).

### Setting Up File Transfer Defaults

Filename mapping defaults and transfer type defaults are defined using the map verb. When a file is transferred from one system to another, if the file specification on a source\_mask parameter matches the name of the file being transferred, AFTP applies the defaults defined in the associated target\_mask and options parameters.

When the AFTP programs search for a match, each source\_mask parameter is read until a match is found. All map verbs that follow the verb containing the match are ignored.

**Note:** The filename mapping defaults are applied by the system *receiving* the file, whether client or server.

File transfer type defaults, which are defined on the options parameter of the map verb, are always applied by the AFTP client, regardless of whether the client is sending or receiving a file.

Following is a description of the MAP verb parameters.

**source\_mask(value)**

The source\_mask parameter is used to specify a file name or file type to map. If the source\_mask matches the name of the file being transferred, the file name mapping defaults defined on the target\_mask and options parameters are applied. The source\_mask parameter value can contain any valid filename character and the wildcard characters asterisk (\*) and question mark (?).

**target\_mask(value)**

The target\_mask is used specify how to name the file or files defined on the source\_mask parameter on the target computer. The target\_mask parameter value can contain any valid filename character and the wildcard characters asterisk (\*) and question mark (?). Characters or wildcards can be enclosed in brackets to indicate that parts of the filename are to be omitted from the file name on the target system.

**options(value)**

The options parameter is optional and is used to specify which file transfer type to use when transferring a file that matches the source\_mask parameter.

These are the values you can use on the options parameter:

**binary** When the options parameter is set to binary, the file is transferred as a stream of binary bytes.

**ascii** When the options parameter is set to ascii, the file is transferred in ASCII.

## Examples

The examples in Table 5 show the various ways file transfer defaults can be used. These examples are based on the following map verbs:

```
map source_mask(*.txt) target_mask(*.text) options (ascii);
map source_mask(*.scr) target_mask(*.script) options (ascii);
map source_mask(*.*bin) target_mask(*.*[bin]) options (binary);
map source_mask(*.*) target_mask(*.???[*]) options (binary);
map source_mask(*.*.c) target_mask([*.*].c) options (binary);
map source_mask(*.*BIN) target_mask(*.*) options (binary);
```

Table 5. File Transfer Default Examples

Source File Name	Source Mask	Target Mask	Mapped File Name
taxplan.txt	*.txt	*.text	taxplan.text
install.scr	*.scr	*.script	install.script
program.exebin	*.*bin	*.*[bin]	program.exe
program.combin	*.*	*.???[*]	program.com
user.program.c	*.*.c	[*.*].c	program.c

Table 5. File Transfer Default Examples (continued)

Source File Name	Source Mask	Target Mask	Mapped File Name
user.plan.script	*.*.script	[*]*.scr	plan.scr
program.zipbin	*.*BIN	*.*	program.zipbin

## AFTP Initialization File Sample

The sample AFTP initialization file is illustrated in Figure 21. The sample can be used on both MVS and VM, and is ready to use. You can tailor the file to meet the needs of your organization.

```
/* AFTP client and server sample initialization file for VM and MVS */
```

```
map source_mask(*.      ) target_mask(*.NONE) options(binary);
map source_mask(*.A37)  target_mask(*.ASSEMBLE) options();
map source_mask(*.BOOK ) target_mask(*.*      ) options(binary);
map source_mask(*.BKINDEX) target_mask(*.*      ) options(binary);
map source_mask(*.LISTPS) target_mask(*.*      ) options(binary);
map source_mask(*.*BIN) target_mask(*.*      ) options(binary);
map source_mask(*.ATF)  target_mask(*.APLTF  ) options(binary);
map source_mask(*.EXC)  target_mask(*.EXEC   ) options();
map source_mask(*.PKG)  target_mask(*.PACKAGE ) options();
map source_mask(*.SCR)  target_mask(*.SCRIPT  ) options(ascii);
map source_mask(*.B00)  target_mask(*.BOOK   ) options(binary);
map source_mask(*.ARC)  target_mask(*.ARCBIN  ) options(binary);
map source_mask(*.BAS)  target_mask(*.BASBIN   ) options(binary);
map source_mask(*.BIN)  target_mask(*.*      ) options(binary);
map source_mask(*.CFG)  target_mask(*.CFGBIN  ) options(binary);
map source_mask(*.COM)  target_mask(*.COMBIN   ) options(binary);
map source_mask(*.EXE)  target_mask(*.EXEBIN   ) options(binary);
map source_mask(*.FLS)  target_mask(*.FLSBIN   ) options(binary);
map source_mask(*.LIB)  target_mask(*.LIBBIN   ) options(binary);
map source_mask(*.OBJ)  target_mask(*.OBJBIN   ) options(binary);
map source_mask(*.PIC)  target_mask(*.PICBIN   ) options(binary);
map source_mask(*.RAM)  target_mask(*.RAMBIN   ) options(binary);
map source_mask(*.SYS)  target_mask(*.SYSBIN   ) options(binary);
map source_mask(*.WKS)  target_mask(*.WKSBIN   ) options(binary);
map source_mask(*.WK1)  target_mask(*.WK1BIN   ) options(binary);
map source_mask(*.WK3)  target_mask(*.WK3BIN   ) options(binary);
map source_mask(*.NEW)  target_mask(*.NEWS    ) options();
map source_mask(*.AVL)  target_mask(*.AVAIL   ) options();
map source_mask(*.BMP)  target_mask(*.BMPBIN   ) options(binary);
map source_mask(*.DLG)  target_mask(*.DLGBIN   ) options(binary);
map source_mask(*.DLL)  target_mask(*.DLLBIN   ) options(binary);
map source_mask(*.ICO)  target_mask(*.ICOBIN   ) options(binary);
map source_mask(*.FNT)  target_mask(*.FNTBIN   ) options(binary);
map source_mask(*.PTR)  target_mask(*.PTRBIN   ) options(binary);
map source_mask(*.RES)  target_mask(*.RESBIN   ) options(binary);
map source_mask(*.ZIP)  target_mask(*.ZIPBIN   ) options(binary);
map source_mask(*.ANN)  target_mask(*.ANNOUNCE) options();
map source_mask(*.DOC)  target_mask(*.DOC     ) options();
map source_mask(*.CAT)  target_mask(*.CATALOG  ) options();
map source_mask(*.IMG)  target_mask(*.IMGBIN   ) options(binary);
map source_mask(*.KPS)  target_mask(*.KPSBIN   ) options(binary);
map source_mask(*.KPL)  target_mask(*.KPLBIN   ) options(binary);
map source_mask(*.PAL)  target_mask(*.PALBIN   ) options(binary);
map source_mask(*.PLX)  target_mask(*.PLXBIN   ) options(binary);
map source_mask(*.R1P)  target_mask(*.R1PBIN   ) options(binary);
map source_mask(*.WRK)  target_mask(*.WRKBIN   ) options(binary);
```

Figure 21. Sample AFTP Initialization File

---

## The ANAME Initialization File

Using the ANAME server initialization file allows you to:

- Define ANAME system administrators.
- Restrict updates to the database.
- Trace the ANAME server.

The ANAME server does not require the initialization file, but if the file is not found, ANAME will use default settings, as follows:

- No system administrator is defined.
- All users can update the database.
- The trace facility is not started when the ANAME database is started.

**Note:** The ANAME client does not use an initialization file.

A sample initialization file is provided for the ANAME server with the APPC Application Suite.

- On VM, when the sample file was installed, it was named APPMANVI INI. The sample should have been renamed to ANAMED INI.
- On MVS, when the sample file was installed, it was named APPMANVI. The sample should have been renamed to APPNAMDI.

If the sample initialization file for your operating system has not been renamed, rename it now.

The sample ANAME initialization file is illustrated in Figure 22 on page 41.

## Syntax

In the ANAMED initialization file, *verbs* are followed by parameter values. Parameter values are enclosed within parentheses. A semi-colon (;) is used after the last parameter value to indicate the end of the verb. Verbs and parameters are *not* case sensitive. The correct verb syntax is:

```
verb_name  
  parameter1(value1)  
  parameter2(value2)  
  .  
  .  
  parameterN(valueN);
```

Blanks, carriage returns, and tabs are ignored. You can use comments in initialization the files (*/\* \*/*), but comments cannot be nested (*/\* /\* \*/*).

## Defining ANAME System Administrators

ANAME system administrators are defined in the ANAME initialization file using the `set_system_administrator` verb in combination with identifiers, which can be either user IDs or fully qualified LU names.

When an administrator is identified by a user ID, the administrator can use ANAME administrator privileges from anywhere in the ANAME network by specifying security flags on ANAME commands. When an administrator is identified by a fully qualified LU name, system administrator privileges are available to anyone using the computer on which the fully qualified LU name is defined.

In the following example, three system administrators are defined:

```
set_system_administrator user_id(user1)
                        user_id(user2)
                        lu_name(netid.luname);
```

**Notes:**

1. When the ANAME server is installed on a VM or an MVS host, the user ID that is used to identify an ANAME administrator must match a valid VM or MVS user ID.
2. If the fully qualified LU name that identifies a system administrator is defined on a MVS or VM host, all users who share that fully qualified LU name have system administrator privileges. Thus, it is strongly recommended that ANAME administrators who use VM or MVS ANAME clients be identified by user IDs.

See “Chapter 6. Managing the ANAME Server” on page 51 for information about ANAME system administrator tasks.

## Preventing Updates

ANAME uses fully qualified LU names as its security key. Any user can add (register) or delete all records associated with his or her own fully qualified LU name. Users are not allowed to register or delete records associated with another user’s fully qualified LU name.

On host systems such as VM and MVS, users share the same fully qualified LU name, so any user can add multiple user names, and can delete any database record. Thus, in a host environment, it is recommended that the ANAME administrator restrict VM and MVS ANAME end users from issuing register and delete commands.

Use the `prevent_update` verb in combination with a fully qualified LU name or names to restrict ANAME end users from issuing register and delete commands. Full and partial wildcards are allowed.

**Note:** Restricted users can still query the database and use the user names defined in the database.

### Examples

If a fully qualified LU name is defined on an MVS or VM host, all users who share that LU name are restricted from updating the ANAME database. If a fully qualified LU name is defined on a workstation, only a user using that workstation is restricted from updating the ANAME database.

In the following example, a fully qualified LU name is specified on the `prevent_update` verb:

```
prevent_update lu_name(NRK321213.D489273);
```

If your organization uses a naming convention for LU names, you can use wildcard characters to restrict multiple LU names. An asterisk (\*) can be used as a full or partial wildcard to represent 0 or more characters and can be placed in any position. For example, if all LU names defined on MVS systems contain MVS in the name, you can restrict all MVS users as follows:

```
prevent_update lu_name(netid.*mvs*);
```

A question mark (?) can be used as a placeholder wildcard. The question mark represents one character in a specific position. For example, you could restrict a group of VM users as follows:

```
prevent_update lu_name (netid.vm?rtp*);
```

You can prevent all users (except ANAME administrators) from updating the ANAME database by specifying a single asterisk (\*). For example:

```
prevent_update 1u_name(*);
```

## Setting Trace for the ANAME Server

To trace the ANAME server, the `set_trace` verb in the ANAME initialization file must be set to a non-zero value. If a support organization needs trace information for the ANAME server to help diagnose a problem, you will be given a trace value for the `set_trace` verb. Enter the trace value provided by support personnel on the `set_trace` verb. Then, stop and restart the ANAME server.

When you finish tracing the ANAME server, either change the value on the `set_trace` verb to zero, or place a comment on the line that contains the `set_trace` verb. Then, stop and restart the ANAME server.

See the *z/OS Communications Server: APPC Application Suite User's Guide* for additional information about tracing the ANAME server.

## ANAME Initialization File Sample

The sample ANAME initialization file is illustrated in Figure 22 on page 41. Before you use the sample file, replace the sample parameter values with actual parameter values.

```

/*****
 * Define the NameServer System Administrator.
 * You can identify ANAME administrators using
 * either user IDs, or fully qualified LU names,
 * or both. ANAME administrators on VM or MVS
 * hosts should be identified by user ID. ANAME
 * administrators on OS/2 workstations can be
 * identified by either a user ID or by the fully
 * qualified LU name defined on their workstation.
 */
    set_system_administrator user_id(UserID1)
                             user_id(UserID2)
                             lu_name(netid.lu1);

/*****
 * Prevent update - specifies fully qualified LU
 * names that are not allowed to register
 * or delete ANAME database records.
 *
 * Because many users share the same fully qualified LU name
 * on host systems like MVS and VM, you should prevent host
 * users from issuing register and delete commands.
 *
 * Multiple fully qualified LU names are allowed.
 * Wildcard characters are allowed.
 */
    prevent_update lu_name(netid.*mvs*)
                  lu_name(netid.abc?vm*);

/*****
 * Change set_trace value if you want the NameServer server to
 * produce a trace file.
 */
    set_trace level( 0 );

```

Figure 22. Sample ANAME Initialization File

---

## The A3270 Initialization Files

Because the A3270 initialization files are different on the VM and MVS operating systems, the initialization file for each operating system is described in a separate section.

Sample initialization files are provided for the A3270 server with the APPC Application Suite.

- On VM, when the sample file was installed, it was named APP3270V INI. The sample should have been renamed to A3270D INI.
- On MVS, when the sample file was installed, it was named APP3270M. The sample should have been renamed to APP3270I.

If the sample initialization file for your operating system has not been renamed, rename it now.

### The VM A3270 Initialization File

On VM, using the A3270 initialization file allows you to:

- Set new defaults for the A3270 set command parameters.
- Set the maximum number of clients that the server will accept.

On VM, the A3270 server does not require the initialization file. If the initialization file is not found, the A3270 will use default settings, as follows:

- Polling is set to 900.
- Inactivity timeout is set to 0.
- Maxclients is set to 500.

### VM A3270 Initialization File Sample

The sample A3270 initialization file for VM is illustrated in Figure 23.

```
/* A3270 sample profile for VM A3270 server
*/

/* Poll clients for activity every 600 seconds (10 minutes) */
set polling 600

/* Disconnect clients that are inactive for 600 seconds (10 minutes) */
set inactivity timeout 600

/* Set the maximum number of clients allowed. */
maxclients 2000

/* End of A3270 server profile for VM */
```

Figure 23. Sample A3270 Virtual Machine Profile

The statements used in the A3270 initialization file are described below.

#### **MAXCLIENTS** *number*

Specifies the maximum number of active terminal sessions that the A3270 server will accept. The valid range is 0 to 4000. When maxclients is set to 0, A3270 clients cannot connect to the server.

The maxclients default is 500. If the maxclients command is omitted from the initialization file, the default is used. The maxclients command can only be issued from the initialization file. If issued from the command line, an error occurs and the command is ignored.

#### **SET INACTIVITY TIMEOUTPOLLING** *n*

Sets either inactivity timeout or polling. Multiple set commands are allowed. The A3270 set command can be issued from either the A3270 initialization file or from the command line. For additional information about the A3270 set command, see “set” on page 74.

## The MVS A3270 Initialization File

On MVS, using the A3270 initialization allows you to:

- Specify the maximum number of clients from which the A3270 server can accept simultaneous connections.
- Define the default application.

You can specify the name of a host application to which A3270 clients are automatically connected. For example, if TSO1 is defined as the default application, clients are connected to the TSO1 application automatically when clients connect to the server without specifying an application name.

- Define LU 2s to the A3270 server

The LU 2s that you list in the A3270 initialization file must be already defined to VTAM in the A3270 APPL. Thus, the LU 2 ACBNames in the initialization file must match the LU 2 ACBNames defined in the A3270 APPL.

- Restrict or allow access to host applications.



## Sample MVS A3270 Initialization File

A sample A3270 initialization file is provided with the APPC Application Suite. The member name of the sample initialization file is APP3270I. The sample initialization file is shown in Figure 24.

```
/******  
* THIS IS THE A3270 SERVER INITIALIZATION FILE  
*  
*****/  
  
/* MAXCLIENTS TELLS THE SERVER HOW MANY CLIENTS TO EXPECT */  
MAXCLIENTS 10  
  
/* SET INACTIVITY TIMEOUT SPECIFIES THE AMOUNT OF TIME */  
/* IN SECONDS AFTER WHICH THE SERVER DISCONNECTS */  
/* INACTIVE CLIENTS. SET INACTIVITY TIMEOUT CAN ALSO */  
/* BE ISSUED FROM THE COMMAND LINE. */  
SET INACTIVITY TIMEOUT 600  
  
BEGINVTAM  
  
/* DEFAULTAPPL IS THE DEFAULT APPLICATION NAME TO WHICH */  
/* CLIENTS WILL GO IF THEY DO NOT EXPLICITLY PROVIDE AN */  
/* APPLICATION NAME. */  
DEFAULTAPPL SAMON  
  
/* ACBNAME IS THE LU TO WHICH ALL CLIENTS SHOULD */  
/* PROVIDE AS THE DESTINATION NAME. */  
ACBNAME A3270D  
  
/* TO USE A CUSTOMER-DEFINED LOGMODE, TYPE THE DEVICE */  
/* NAME AND LOGMODE NAME. THE SYNTAX IS */  
/* DEVICE_NAME LOGMODE_NAME. */  
IBM-3278-2 LOGMODE1  
  
/* ALLOWAPPL AND RESTRICTAPPL STATEMENTS DETERMINE THE */  
/* APPLICATIONS TO WHICH CLIENTS HAVE ACCESS. */  
/* "ALLOWAPPL *" ALLOWS CLIENTS TO CONNECT TO ANY */  
/* APPLICATION. */  
ALLOWAPPL *  
  
/* DEFAULTTLUS INDICATES THE BEGINNING OF THE LU2 APPL */  
/* MINOR NODE LIST. */  
DEFAULTTLUS  
A32T01 A32T02 A32T03 A32T04 A32T05 A32T06 A32T07 A32T08  
A32T09 A32T10  
ENDEFAULTLUS  
ENDVTAM
```

Figure 24. Sample A3270 Initialization File

The statements used in the A3270 initialization file are described below.

### **MAXCLIENTS** *number*

Specifies the maximum number of clients from which the A3270 server can accept simultaneous connections. The valid range is 1 to 32 000.

This statement must be *before* the BEGINVTAM statement or *after* the ENDVTAM statement.

### **SET INACTIVITY TIMEOUTPOLLING** *n*

Sets either inactivity timeout or polling. Multiple set commands are allowed. The

A3270 set command can be issued from either the A3270 initialization file or from the command line. For additional information about the A3270 set command, see “set” on page 74.

This statement must be *before* the BEGINVTAM statement or *after* the ENDVTAM statement.

### **BEGINVTAM**

Indicates the beginning of the list of VTAM parameters for the A3270 server.

### **DEFAULTAPPL**

Defines the host application to which to connect when an A3270 client connects to the server without specifying an application name. This statement must be placed *between* the BEGINVTAM and ENDVTAM statements.

**Note:** The DEFAULTAPPL statement is not required. However, if a default host application is not specified, the A3270 client must always supply an application name when connecting to the server.

### **DEFAULTLUS**

Indicates the beginning of the list of the LU 2 ACBNAMEs that are available for this server. The LU 2 ACBNAMEs following the DEFAULTLUS parameter must match the LU 2 ACBNAMEs defined to VTAM in the A3270 server’s VTAM application program major node definition. This statement must be placed *between* the BEGINVTAM and ENDVTAM statements.

**Note:** Multiple instances of the A3270 server running in the same host cannot share LU 2s. Thus, the LU 2 ACBNAMEs for each instance of the server must be unique.

### **ENDDEFAULTLUS**

Ends the DEFAULTLUS list. This statement must follow the DEFAULTLUS statement.

### **ACBNAME**

Contains the ACBNAME name of the A3270 server. The ACBNAME must match the ACBNAME name defined in the application program major node. The ACBNAME statement is required and must be placed *between* the BEGINVTAM and ENDVTAM statements.

#### *device\_name logmode*

Specifies a customer-defined logmode. If customer-defined logmodes are not specified, VTAM default logmodes are used. See “Appendix B. A3270 Default Logmodes” on page 83 for a list of supported device names and the corresponding VTAM default logmodes.

### **RESTRICTAPPL**

Specifies the names of specific host applications that A3270 clients are *not* permitted to access. Asterisks in an application name represent any group of characters of any length. A question mark represents a single character.

If an A3270 client supplies an application name that is restricted, the A3270 server rejects the restricted application name and asks the client to supply a different application name. The RESTRICTAPPL statement is not required.

**Note:** RESTRICTAPPL \* restricts access to *all* host applications except the default application. The default application cannot be restricted.

### **ALLOWAPPL**

This parameter specifies the names of applications that A3270 clients are permitted to access. Trailing asterisks in an application name represent any

group of characters of any length. A question mark represents a single character. The ALLOWAPPL parameter is not required. It must follow the BEGINVTAM statement, and must precede the ENDVTAM statement.

**Notes:**

1. Although the ALLOWAPPL statement is not required, if the statement is omitted, the A3270 rejects all application names provided by A3270 clients, so all attempts to connect to the server will fail.
2. If you define a default application (DEFAULTAPPL), the default application cannot be restricted. Thus, the default application is always allowed.
3. When determining whether an A3270 client can access a specific host application, the A3270 server first checks the list of restricted applications (RESTRICTAPPL). If the application specified by the client is found, the connection is rejected, and the list of allowed applications is not checked. Otherwise, the server checks the list of allowed applications (ALLOWAPPL). If a match is found, the connection is accepted. If a match is not found on either the restricted or the allowed list, the connection is rejected.

If the following statements appear in the initialization file, an application named ECHO02A would be rejected.

```
RESTRICTAPPL ECHO*  
ALLOWAPPL *
```

If the following statements appear in the initialization file, *all* host applications other than the default application would be restricted.

```
RESTRICTAPPL *  
ALLOWAPPL *
```

**ENDVTAM**

Ends the list of the VTAM parameters.

**Using the APP3INME Utility to Create the A3270 Initialization File**

The APP3INME utility, which is supplied with the APPC Application Suite, is a TSO/REXX EXEC that creates an initialization file for the MVS A3270 server. By specifying a few parameters, you can create the initialization file from your TSO ID.

To use the APP3INME utility, type:

```
EXEC 'name(APP3INME)' 'appc_acb lu2_acb_prefix num_lu2s default_app1'
```

where:

- name is the name of the partitioned dataset (PDS) in which the APPEINME EXEC is installed.
- appc\_acb is the A3270 server's ACB name.
- lu2\_acb\_prefix is LU 2 ACB name prefix of the LU 2s that are defined in the A3270 application program major node.
- num\_lu2s is the number of LU 2s that are defined in the A3270 application program major node.
- default\_app1 is the name of the default application.

The APP3INME utility creates the initialization file in a PDS named *your\_id.vtamlst(APP3INME)*, where *your\_id* is your TSO user ID. You can copy the initialization file from *your\_id.vtamlst* to any partitioned dataset.

For example, if you enter:

```
EXEC 'YOURID.CLIST(APP3INME)' 'A3270D A32T 10 SAMON'
```

APP3INME creates an A3270 initialization file with a member name of APP3270I which contains:

- The A3270 server's ACB name, which is A3270D.
- 10 LU 2 ACBNAMEs that contain a prefix of **A32T** and are numbered 1 to 10.
- An ALLOWAPPL \* statement.
- A default application named SAMOM.

**Note:** You might want to customize the ALLOWAPPL statement created by the APP3INME utility. Additionally, you can add a RESTRICTAPPL statement.

---

## Chapter 5. Managing AFTP Servers

When the AFTP server receives one of the following commands from an AFTP client, the server enters a record in the AFTP audit log:

- acopy
- get
- put
- rename
- delete
- mkdir
- rmdir

The audit log can be particularly useful when you need to determine the identity of end users who have requested unauthorized actions. You can also use the audit log to assist end users who are having difficulty accessing needed files or directories.

An example of an AFTP server audit log is shown in Figure 25 on page 48.

```

-----
Audit Log For Program      : "AFTPD" AFTP Server Program
Program Version Number    : n.nn
Program Log Entry Time    : Thu May 05 16:16:18 1994

Userid Attempting Command : ANONYMOUS
FQLU Name of Partner      : USIBMNR.NR50F241
Current Working Directory : /E:/WORK/AFTP/MAKE/TESTINI/SERVER/
Command Issued            :
PUT "/E:/WORK/AFTP/MAKE/TESTINI/SERVER/test.dat".
Command Completed Successfully.

-----
Audit Log For Program      : "AFTPD" AFTP Server Program
Program Version Number    : n.nn
Program Log Entry Time    : Thu May 05 16:16:24 1994

Userid Attempting Command : LBONANNO
FQLU Name of Partner      : USIBMNR.NR50FF8I
Current Working Directory : /E:/WORK/AFTP/MAKE/TESTINI/SERVER/
Command Issued            :
PUT "/E:/WORK/AFTP/MAKE/TESTINI/SERVER/ZERO.DAT"
Command Completed Successfully.

-----
Audit Log For Program      : "AFTPD" AFTP Server Program
Program Version Number    : n.nn
Program Log Entry Time    : Thu May 05 16:16:28 1994

Userid Attempting Command : LBONANNO
FQLU Name of Partner      : USIBMNR.NR50FF8I
Current Working Directory : /E:/WORK/AFTP/MAKE/TESTINI/SERVER/
Command Issued            :
DELETE "/E:/WORK/AFTP/MAKE/TESTINI/SERVER/ZERO.DAT"
Command Failed.

-----
Audit Log For Program      : "AFTPD" AFTP Server Program
Program Version Number    : n.nn
Program Log Entry Time    : Thu May 05 16:16:34 1994

Userid Attempting Command : LBONANNO
FQLU Name of Partner      : USIBMNR.NR50FF8I
Current Working Directory : /E:/WORK/AFTP/MAKE/TESTINI/SERVER/
Command Issued            :
MKDIR "test"
Command Completed Successfully.

-----

```

*Figure 25. Sample AFTP Server Audit Log*

The fields in the AFTPD audit log are described below:

**Audit Log For Program**

The name of the program for which the log entry was made.

**Program Version Number**

The version of the program for which the log entry was made.

**Program Log Entry Time**

The date and time the log entry was made.

**Userid Attempting Command**

The user ID of the end user who issued the command recorded in the log entry.

**FQLU Name of Partner**

The fully qualified LU name of the client that issued the command recorded in the log entry.

**Current Working Directory**

The name of the server's current working directory when the command recorded in the log entry was issued.

**Command Issued**

The name of the command entered by the AFTP client, followed by a message indicating if the command completed successfully.





---

## Chapter 6. Managing the ANAME Server

The APPC NameServer (ANAME) provides a global naming facility that allows you to easily identify the name of a partner in a computer network. ANAME provides three types of naming services:

- ANAME eliminates the need for users to know LU names. LU names are mapped to more useable nicknames (user names) that are accessible across the network. The user can use these nicknames in place of LU names. For example, rather than trying to remember a fully qualified LU name like NETID.ABC11111, you can use a simple nickname like PrintServer.
- In addition to providing a global naming facility, the group membership function of ANAME allows LUs to be categorized according to common function or group association. This assists in locating specific services or in identifying characteristics of an LU. For example, all LUs that serve as global gateways can be identified by registering as a member of a group named Gateways.
- An additional use of ANAME is to identify transaction programs that reside on particular LUs. Certain transaction programs require a server that is shared by all users in the network. In many cases, the server for this transaction program must be defined on every workstation using the service. ANAME allows these network-wide servers to be identified by registering specific transaction programs with a LU in the ANAME database. This eliminates the complexity of reconfiguration when the transaction program server is relocated.

The APPC NameServer consists of two transaction programs: ANAME, which is the client program and ANAMED, which is the server program.

For additional information about ANAME, see the *z/OS Communications Server: APPC Application Suite User's Guide*.

---

### Starting and Stopping the ANAME Server

This section describes how to start and stop the ANAME server on both the VM and MVS operating systems.

#### Starting the Server on VM

On VM, you can start the ANAME server by issuing the `anamed` command on the server's virtual machine. The server will remain active until it is stopped using the `halt` command, or until AVS or VTAM are shut down. See "anamed" on page 61 for `anamed` command syntax.

#### Example

The following example illustrates how the ANAME server is started with the `ANAMED` command on VM.

```
anamed
IBM APPC NameServer version 1.0.. Copyright (C) 1994 by IBM
All rights reserved.
```

```
Opening initialization file from anamed.ini.
APP0749E The initialization file was successfully loaded.
```

## Starting the Server Automatically

You can start the server automatically by placing the `anamed` command in the server's logon profile. Additionally, it is recommended that you define the ANAME server's virtual machine to your system automation so the server's virtual machine is started automatically.

For additional information about the `anamed` command, see “`anamed`” on page 61.

For additional information about the server's logon profile, see “Create a Logon Profile” on page 10.

## Stopping the Server on VM

Stop the ANAME server by entering the `halt` command on the ANAME server's virtual machine console. If the server is processing a request, the server will stop when it has finished that request. Otherwise, the server stops immediately. See “`halt`” on page 62 for `halt` command syntax.

### Example

```
halt
Shutdown accepted. The nameserver will exit shortly.
```

## Starting the Server on MVS

On MVS, the APPC attach manager starts the ANAME server automatically when the first request is received from an ANAME client. The server is configured to be a long running transaction; however, any of the following will cause the ANAME server to shut down:

- The `halt` command is issued.
- The server is inactive for five minutes.
- APPC/MVS is shut down.

## Stopping the Server on MVS

To stop the ANAME server, modify the ANAME TP profile JCL using this procedure:

1. Change the `TPADD` utility command to `TPMODIFY`.
2. Change the `ACTIVE` keyword from `ACTIVE(YES)` to `ACTIVE(NO)`.
3. Run the JCL.

If the ANAME server does not stop within six minutes, follow this procedure: use the APPC/MVS TP display command to determine whether ANAME clients are using the server. The APPC/MVS TP display command syntax is:

```
DISPLAY APPC,TP,ALL
```

If no ANAME clients are using the server, cancel the `ANAMED` job.

**Note:** When `ACTIVE(YES)` is changed to `ACTIVE(NO)`, the APPC attach manager is not able to start the ANAME server. Thus, before the ANAME server can be started again, you must change `ACTIVE(NO)` to `ACTIVE(YES)`.

---

## ANAME Database Administration

This section describes the tasks performed by the ANAME system administrator (ANAME administrator), which are:

- Defining ANAME administrators.
- Managing records for other users.
- Adding duplicate user names.

- Backing up the ANAME database.
- Recovering the ANAME database.

## Defining ANAME Administrators

ANAME administrators are defined to ANAME in the server's initialization file and are identified either by a fully qualified LU name, or a user ID.

When an administrator is identified by a user ID, the administrator can use ANAME administrator privileges from anywhere in the ANAME network by specifying security flags on certain ANAME commands. When an administrator is identified by a fully qualified LU name, system administrator privileges are available to anyone using the computer on which the fully qualified LU name is defined.

### Notes:

1. When the ANAME server is installed on a VM or MVS host, the user ID that is used to identify an ANAME administrator in the ANAME initialization file must be a valid VM or MVS user ID.
2. If the fully qualified LU name that identifies a system administrator is defined on a MVS or VM host, all users who share that fully qualified LU name have system administrator privileges. Thus, it is strongly recommended that ANAME administrators who use ANAME clients on MVS or VM be identified by user IDs.

## Using ANAME Administrator Security

If an ANAME administrator is identified by a user ID, the ANAME administrator must specify security flags on the ANAME register and delete commands to add or delete records for other users, or to add duplicate user names. If security flags are not used when required, an error occurs and the ANAME command fails.

If the ANAME server is installed on a VM or MVS host, use the information that follows to help you determine which security flags to use.

- The `-c` flag indicates APPC security=same. When you use the `-c` flag (security=same), APPC obtains the user ID and password that are configured on the ANAME client system. APPC sends the client user ID and password to the ANAME server system (VM or MVS). If the user ID and password obtained from the client system is a valid VM or MVS user ID and password, APPC passes the user ID to the ANAME server for comparison with the user ID defined in the ANAME initialization file. Thus, to use the `-c` flag, the user ID and password that are configured on the ANAME client system must be a valid VM or MVS user ID and password, and the user ID must match the user ID defined in the ANAME initialization file.

**Note:** For additional information about using APPC security=same, see “Appendix C. Configuring APPC to Support APPC Security SAME” on page 85.

- The `-a` (user ID) flag and `-p` (password) flag indicate APPC security=program. The `-a` and `-p` flags allow you to specify an explicit user ID and password. If the user ID and password specified with the `-a` and `-p` flags match a valid VM or MVS user ID and password, APPC passes the user ID to the ANAME server for comparison to the user ID defined in the ANAME initialization file. Thus, the user ID and password specified with the `-a` and `-p` flags must match a valid VM or MVS user ID and password, and the user ID must match the user ID defined in the ANAME initialization file.

See “ANAME Administrator’s Command Reference” on page 56 for information about register and delete command syntax.

## Managing Records for Other Users

Although end users can only register or delete ANAME database entries that contain their own fully qualified LU name, ANAME allows ANAME administrators to register (add) or delete any entry. ANAME administrators can use the fully qualified LU name parameter (-f) with either the ANAME register or ANAME delete command to register or delete entries for other users.

See “ANAME Administrator’s Command Reference” on page 56 for complete register and delete command syntax.

**Note:** ANAME administrators do not need to use the fully qualified LU name parameter when registering or deleting records that contain their own fully qualified LU name.

### Examples

- In this example, the ANAME administrator adds a record for another user. The user’s user name is Bodine, and the user’s fully qualified LU name is LU.A.  

```
ANAME REG -u Bodine -f LU.A
```
- The system administrator can delete the same record:  

```
ANAME DEL -u Bodine -f LU.A
```
- The system administrator can delete all records in the database with one delete command:  

```
ANAME DEL -f *
```

## Adding Duplicate User Names

In most cases, user names are unique within the ANAME network and are associated with a single fully qualified LU name. However, some systems have more than one LU name, and you might want to register the same user name for each of these LU names. Thus, ANAME allows system administrators to register duplicate user names. A duplicate user name is a user name that is associated with more than one fully qualified LU name.

The duplicate flag can be used on the register command to register duplicate user names.

See “ANAME Administrator’s Command Reference” on page 56 for complete register command syntax.

### Examples

The following examples illustrate how administrators register duplicate user names using the -d flag. For the examples, assume this record is contained in the ANAME database:

User Name	Fully Qualified LU Name	Group Name	TP Name
VMSysA	LU.A		

- Register VMSysA with a LU name that is different from LU.A:  

```
ANAME REG -u VMSysA -f LU.B -d
```

The following record is added to the database:

VMSysA	LU.B		
--------	------	--	--

- The transaction program named AFTPD is located on VMSysA and is accessed through LU.C.  

```
ANAME REG -u VMSysA -f LU.C -t AFTPD -d
```

The following record is added to the database:

VMSysA	LU.C	AFTPD
--------	------	-------

An ANAME query of the name VMSysA would now return these records.

User Name	Fully Qualified LU Name	Group Name	TP Name
VMSysA	LU.A		
VMSysA	LU.B		
VMSysA	LU.C		AFTPD

For additional information about using duplicate user names, see “Defining User Names for Host Systems”.

### Defining User Names for Host Systems

ANAME can be used to define user names for all APPC platforms in a network, eliminating the need for users (or programs) to know each platform’s SNA APPC LU name. In the following example, user names are defined for three different APPC platforms.

User Name	LU Name	Group Name	TP Name
RALVM6	NETID.VMSYS	VM	
RALAIIX	NETID.AIXSYS	AIX	
RALMVS	NETID.MVSSYS	MVS	

An MVS or VM host can have more than one LU name. In the following example, a duplicate user name (RALVM6) is defined for a single VM host that has three different LU names.

User Name	LU Name	Group Name	TP Name
RALVM6	NETID.VMSYS1A	VM	
RALVM6	NETID.VMSYS1B	VM	
RALVM6	NETID.VMSYS1C	VM	

## Backing Up the ANAME Database

Because the ANAME server does not maintain a permanent backup copy of the ANAME database file, it is recommended that you create a backup of the ANAME database periodically.

The file name of the ANAME database on VM is ANAMED DB. The DD name of the ANAME database dataset on MVS is APPNAMDF.

**Note:** On VM, ANAME creates a temporary backup copy of the database file when updating the database. However, the temporary backup copy is erased when the database is updated successfully.

## Recovering the ANAME Database

The ANAME does not recover the database file if it is corrupted or lost. You must restore the database if a backup is available.

If the ANAME database is corrupted or lost and a backup file is not available, refer to the instructions for your operating system below.

### VM

Erase the corrupted database if it exists, and register all user names again. ANAME will automatically create a new database when the first register request is received.

## **MVS**

Re-install the seed database file that was shipped with ANAME. Copy the file into the APPNAMDF dataset and overwrite the existing corrupted database if it exists.

If you cannot locate the seed database file, contact IBM Software Support Center.

---

## **ANAME Administrator's Command Reference**

This section documents commands that have special parameters for use by ANAME administrators, and commands that are used to stop and start the ANAME server on VM. See the *z/OS Communications Server: APPC Application Suite User's Guide* for the complete set of ANAME commands.

## aname delete

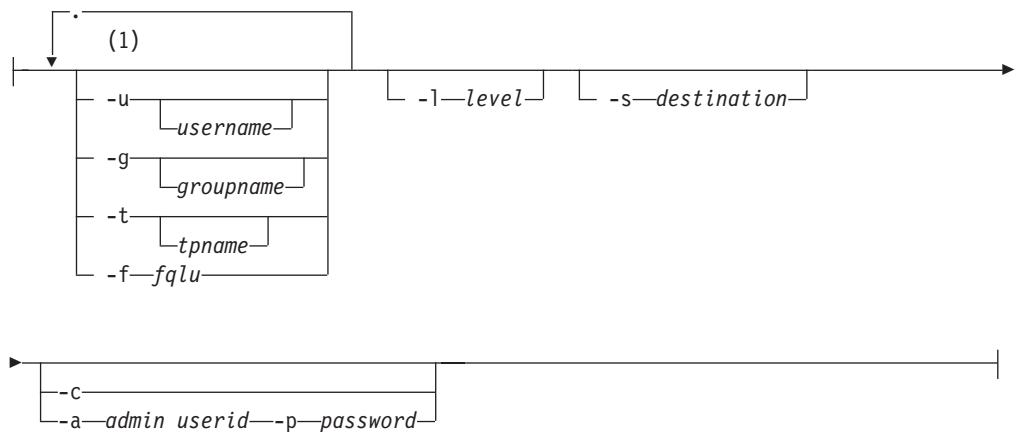
### Purpose

Use this command to delete one or more entries from the database. ANAME administrators can delete records for any user in the ANAME network.

### Format



### parameters:



### Notes:

- 1 You must choose one or more parameters from this group, but may choose each parameter only once.

### Parameters

For a description of remaining parameters, see the *z/OS Communications Server: APPC Application Suite Programming*

- a User ID; the user ID that is defined for an ANAME administrator in the ANAME server's initialization file. If you specify a user ID, a password is also required (-p).
- c Security=same; used when an ANAME administrator's user ID and password on the ANAME client computer match the ANAME administrator's user ID and password on the ANAME server computer. If -c fails, a message is displayed stating that conversation security is not valid. If this error occurs, use the -a flag and -p flag to specify an explicit user ID and password.
- f Fully qualified LU name; the fully qualified LU name of a user other than an ANAME administrator. Used when an ANAME administrator is deleting a record for another user.

See "Managing Records for Other Users" on page 54 for additional information about deleting records for other users.

- p Password; the ANAME administrator's VM or MVS password. If you specify a

password, a user ID is also required (-a). See “Defining ANAME Administrators” on page 53 for information about defining ANAME administrators.



## aname register

### Purpose

Use this command to add a record to the database. ANAME administrators can add records for any user in the ANAME network.

### Notes:

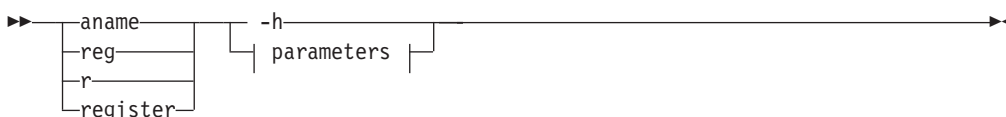
1. If you try to register a record with the same fields as another record already in the database (although the case may vary), the second record will replace the first. For example, if you enter the following two commands:

```
aname reg -u ellenm
aname reg -u EllenM
```

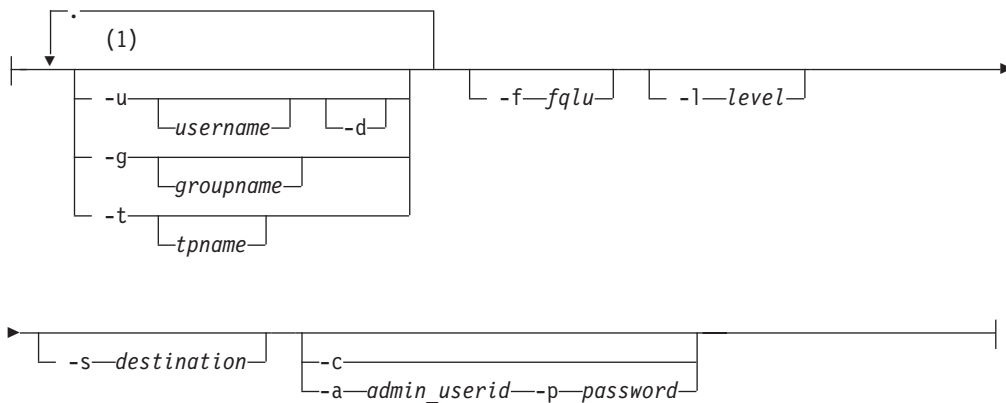
Only one record will actually be in the database:

```
Username Fully Qualified LU
EllenM NET1.LUSVT
```

### Format



### parameters:



### Notes:

- 1 You must choose one or more parameters from this group, but may choose each parameter only once.

### Parameters

For a description of remaining parameters, see the *z/OS Communications Server: APPC Application Suite Programming*

- a User ID; the user ID that is defined for an ANAME administrator in the ANAME server's initialization file. If you specify a user ID, a password is also required (-p).
- c Security=same; used when an ANAME administrator's user ID and password on the ANAME client computer match the ANAME administrator's user ID and

password on the ANAME server computer. If -c fails, a message is displayed stating that conversation security is not valid. If this error occurs, use the -a flag and -p flag to specify an explicit user ID and password.

- d Duplicate; this flag enables an ANAME administrator to register a user name that is associated with more than one fully qualified LU names. See “Adding Duplicate User Names” on page 54 for additional information about adding duplicate user names.
- f Fully qualified LU name; the fully qualified LU name of a user other than an ANAME administrator. Used when an ANAME administrator is deleting a record for another user.  
  
See “Managing Records for Other Users” on page 54 for additional information about deleting records for other users.
- p Password; the ANAME administrator’s VM or MVS password. If you specify a password, a user ID is also required (-a). See “Defining ANAME Administrators” on page 53 for information about defining ANAME administrators.

## anamed

### Purpose

On VM, use this command to start the ANAME server. The anamed command can either be entered on the server's virtual machine console or placed in the server's logon profile. When the anamed command is placed in the server's logon profile, the ANAME server is started automatically.

See "Starting the Server on VM" on page 51 for additional information about starting the ANAME server on VM.

See "Starting the Server on MVS" on page 52 for information about starting the ANAME server on MVS.

### Format

►►—anamed—

---

◄◄

### Parameters

None.

# halt

## Purpose

On VM, use this command to stop the ANAME server.

See “Stopping the Server on VM” on page 52 for additional information about stopping the ANAME server on VM.

See “Stopping the Server on MVS” on page 52 for information about stopping the ANAME server on MVS.

## Format

▶▶ `halt`  ▶▶  
    `ha`

## Parameters

None.

---

## Chapter 7. Managing the A3270 Emulator Server

The A3270 Emulator Server (A3270 server) provides simple 3270 connectivity for VM and MVS users. An A3270 user can have multiple 3270 sessions to one or more A3270 servers. Instead of using an LU type 2 (LU 2) session to transport 3270 data between the client and server, the A3270 server uses APPC conversations. The A3270 server sends only the 3270 data and does not encapsulate the LU 2 data.

The A3270 server:

- Is modeled after TN3270 from the TCP/IP protocol suite.
- Supports applications that rely on CLSDST PASS to provide session manager support.
- Reduces the number of protocols in your network and on your client workstations. Because the number of protocols required on your client workstations are reduced, memory requirements might also be reduced.
- Provides a feature called *RH and sequence number support* to prevent synchronization problems.

The A3270 server supports many terminal types:

- IBM 3277 Model 2, base and extended datastream.
- IBM 3278 Model 2, base and extended datastream.
- IBM 3278 Model 3, base and extended datastream.
- IBM 3278 Model 4, base and extended datastream.
- IBM 3278 Model 5, base and extended datastream.
- IBM 3279 Model 2, base and extended datastream.
- IBM 3279 Model 3, base and extended datastream.
- IBM 3279 Model 4, base and extended datastream.
- IBM 3279 Model 5, base and extended datastream.

A3270 client emulators are available from IBM and other vendors.

---

### How A3270 Works

LU type 2 3270 is a full duplex protocol. Because APPC conversations are half duplex, A3270 clients establish two APPC conversations with the A3270 server. The client uses one conversation to send data and the other to receive data.

A3270 server connections are established after the following events complete successfully:

1. The A3270 client starts an APPC conversation with the A3270 server.
2. The client starts a second APPC conversation with the A3270 server.
3. The client and server negotiate options, such as:
  - Terminal type.
  - Application name (MVS server only).
  - User data, such as the user's user ID (MVS server only).
4. The server opens a logical terminal on behalf of the client:
  - In MVS, the logical terminal is a LU 2 session.
  - In VM, the logical terminal is a CP Logical Device Support Facility (LDSF) session.
5. The client and server exchange 3270 data.

## Using the A3270 Server in Your Network

This section describes the role of the A3270 server in various types of networks.

### A Single Local Host System

The simplest network consists of a single host system and one client.

In Figure 26, an A3270 client is logged on to a virtual machine. First, the client established two APPC conversations with the A3270 server and negotiated options such as terminal type. The A3270 server opened a logical terminal by using the CMS Logical Device Support Facility (LDSF). The A3270 user now has the VM logo on the emulation screen and logs on to the virtual machine.

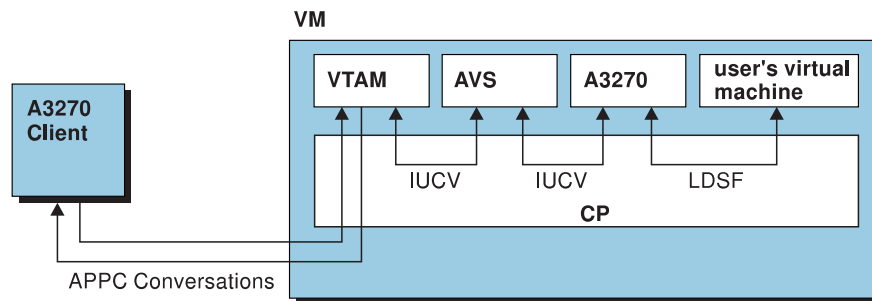


Figure 26. Simple VM Example

In Figure 27, an A3270 client is logged on to an MVS application called CICS\*. First, the client established a pair of APPC conversations with the A3270 server and negotiated options such as application name (CICS), user data, and terminal type. The A3270 server opened a logical terminal by starting an LU 2 session to CICS. The A3270 user now has a CICS logon screen.

**Note:** CICS could be any host application.

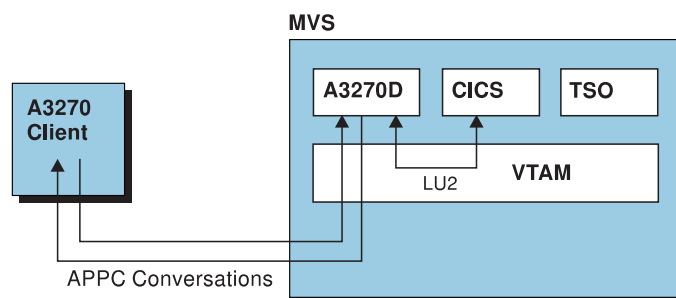


Figure 27. Simple MVS Example

### Multiple Local Host Systems

Figure 28 on page 65 illustrates a network with two MVS systems and two VM systems. In this network, you could install the A3270 server in one MVS system and allow this server to provide A3270 connectivity to applications in the other three host systems. Thus, a user who wants to logon to VMA would connect to the A3270 server and specify VMA as the desired application. The user could start a second A3270 session to VMA and start other sessions to CICS, TSO, or VMB.

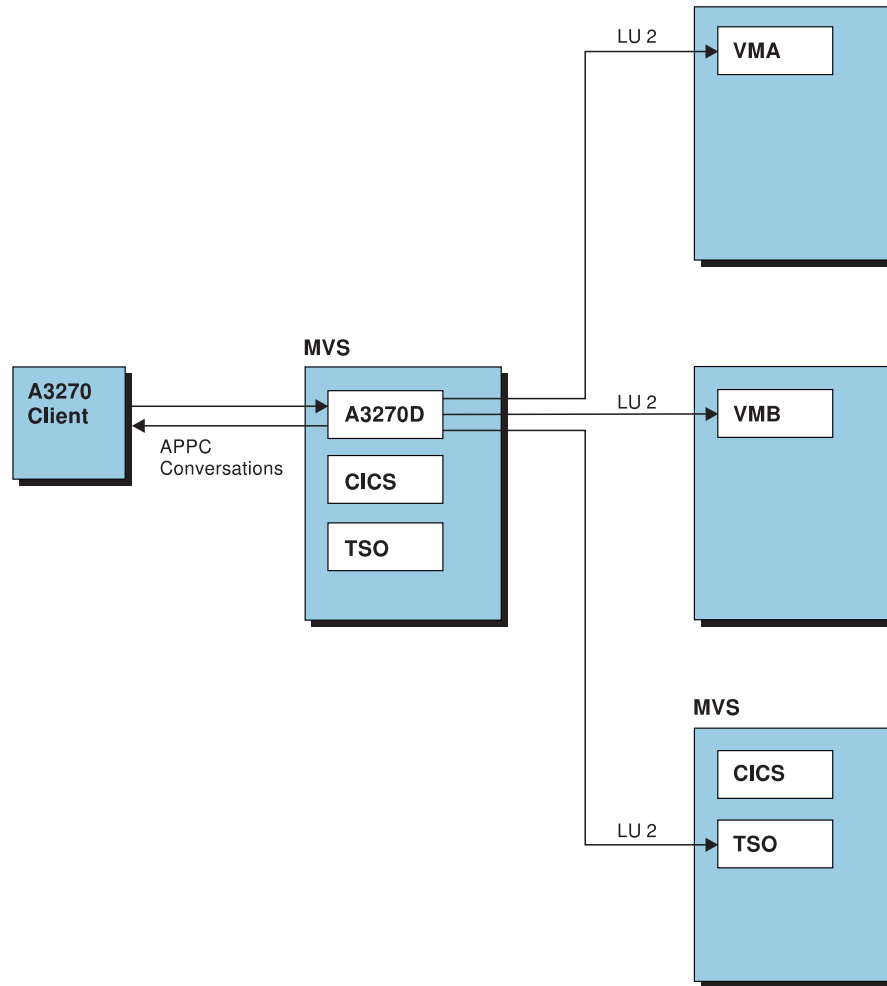


Figure 28. A Complex Network

A VM system with a logon manager installed (such as pass-through virtual machine (PVM)) can be used as an A3270 gateway to allow users to connect to host applications in other systems. An A3270 installed on this type of VM machine can provide the same services as the MVS A3270 server illustrated in Figure 28.

When deciding whether to install an A3270 server on every host or to install the server on selected hosts, the advantages of each should be considered. In many networks, it is preferable to have an A3270 server running on each host. This solution allows the A3270 APPC sessions to be routed directly to the host application, reducing the path length between the A3270 client and the desired host application and often resulting in faster response time for the end user. In contrast, in networks in which A3270 connections must traverse many intermediate nodes, fewer session resources will be used when the APPC portion of the connection is minimized.

---

## Starting and Stopping the A3270 Server

This section describes how to start and stop the A3270 server on both the VM and MVS operating systems.

## Starting the Server on VM

On VM, you can start the A3270 server by issuing the a3270 command on the server's virtual machine. The server will remain active until it is stopped using the halt command, or until AVS or VTAM are shut down. See "a3270" on page 68 for a3270 command syntax.

### Example

The following example illustrates how the A3270 server is started with the a3270 command on VM.

```
a3270
A3270 terminal server version 1.0 starting.
Licensed Materials - Property of IBM
5654-025
Copyright (C) 1994 by IBM Corp. All rights reserved.
The A3270 initialization file DD:APP3270I
was opened successfully.
A3270 server is ready to accept clients.
```

### Starting the Server Automatically

You can start the server automatically by placing the a3270 command in the server's logon profile. Additionally, it is recommended that you define the A3270 server's virtual machine to your system automation so the server's virtual machine is started automatically.

For additional information about the a3270 command, see "a3270" on page 68.

For additional information about the server's logon profile, see "Create a Logon Profile" on page 12.

## Stopping the Server on VM

Stop the A3270 server by entering the halt command on the A3270 server's virtual machine console. See "halt" on page 71 for information about the halt command.

## Starting the Server on MVS

On MVS, start the A3270 server by starting the A3270 server's MVS PROC. The A3270 server's MVS PROC is named APP3270D.

## Stopping the Server on MVS

Stop the A3270 server by entering the halt command. See "halt" on page 71 for information about the halt command.

---

## A3270 Server Command Reference

The A3270 server operator commands allow the operator to:

- Start the A3270 server on VM.
- Stop the A3270 server.
- Display active clients or server statistics.
- Modify certain server parameters.
- Terminate sessions to client.
- Run A3270 server internal tests.

On VM, you can enter A3270 commands from the A3270 virtual machine console. If you prefer to send commands to the A3270 server without logging on to the server's virtual machine, you can set up a secondary operator.



On MVS, A3270 commands are entered using the MVS modify command.

The A3270 set command can also be used in the A3270 server initialization file.

## a3270

### **Purpose**

On VM, use this command to start the A3270 server.

See “Starting the Server on VM” on page 66 for additional information about starting the A3270 server on VM.

See “Starting the Server on MVS” on page 66 for information about starting the A3270 server on MVS.

### **Format**

▶▶ a3270 ◀◀

### **Usage**

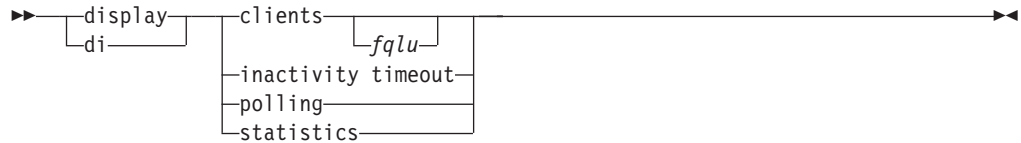
This command can either be entered on the server’s virtual machine console or placed in the server’s logon profile. When the a3270 command is placed in the server’s logon profile, the A3270 server is started automatically.

# display

## Purpose

Use this command to display A3270 server options.

## Format



## Parameters

### clients *fqlu*

Displays information about the A3270 clients that are currently connected to the A3270 server. If the *fqlu* option is entered, the A3270 server displays information about the clients that are connected from the specified fully qualified LU name. Otherwise, information about all connected clients is displayed. This information includes:

#### LDSF ID (VM Only)

The Logical Device Support Facility ID (LDSF ID) used for the connection between this A3270 client and the A3270 server. VM generates the LDSFID automatically.

#### LU Name (MVS Only)

The ACBNAME of the LU 2 used for the connection between this A3270 client and the A3270 server. LU 2 ACBNAMEs are defined in the server's VTAM program application major node.

#### Partner LU Name

The LU name of the system on which the A3270 client emulator is running. The fully qualified LU name is displayed.

#### Read path

The conversation ID of the APPC conversation being used for reading data from the client. The conversation ID is displayed in hexadecimal.

#### Write path

The conversation ID of the APPC conversation being used for writing data to the client. The conversation ID is displayed in hexadecimal.

**State** The state of the client. Possible states are:

- Active - this client is active.
- Terminating - the A3270 server is ending the client's connection.
- Negotiating - a connection is established, but the client and server are still negotiating startup parameters.
- Set-up in progress - a connection is established, but only one of the two required conversations is started.

Acceptable abbreviation: di c1

### inactivity timeout

Displays the current value of the inactivity timeout parameter. For additional information about the inactivity timeout parameter, see "set" on page 74.

Acceptable abbreviation: di in ti

### polling

Displays the current value of the polling parameter. For additional information about the polling parameter, see “set” on page 74.

Acceptable abbreviation: di po

### statistics

Displays A3270 server statistics.

Acceptable abbreviation: di st

## Usage

Use this command from the command line only; if used in the A3270 server’s initialization file, an error will occur.

## Example

- Display clients - VM server

```
di cl
LDSF ID   Partner LU Name   Read path   Write path   State
-----
L003     netid.luA                0             1   Active
L002     netid.luB                4             3   Terminating
```

- Display clients - MVS server

```
di cl
LU NAME   Partner LU Name   Read path   Write path   State
-----
A32T0001 netid.luA                0             1   Active
A32T0002 netid.luB                4             3   Terminating
```

- Display inactivity timeout. In this example, the inactivity timeout parameter is set to zero.

```
di in ti
Inactive clients will not be terminated.
```

In this example, the inactivity timeout parameter is set to 600 seconds.

```
di in ti
Inactive clients will be terminated after 600 seconds.
```

- Display polling. In this example, the polling parameter is set to 600 seconds.

```
di po
Clients will be polled every 600 seconds.
```

- Display statistics.

```
di st
A3270 server version n.nn
The server has been running for 5 hours 9 minutes 18 seconds
Current clients being served:           627
Total number of clients served:         15278
Maximum number of clients served:       976
Clients active:                          626
Clients being terminated:                0
Clients performing negotiation:          1
Clients performing setup:                0
Total data blocks received from clients: 978278
Data blocks sent to clients:             932892
```

# halt

## Purpose

Use this command to stop the A3270 server.

See “Stopping the Server on VM” on page 66 for additional information about stopping the A3270 server on VM.

See “Stopping the Server on MVS” on page 66 for information about stopping the A3270 server on MVS.

## Format



## Parameters

### normal

Stops the A3270 server after all clients have disconnected normally. When `halt` or `halt normal` is issued, the server rejects new connection requests from A3270 clients.

Accepted abbreviation: `ha`.

### quick

Stops the A3270 server after the server sends a termination message to each connected client.

Accepted abbreviation: `ha qu`.

### cancel

Stops the A3270 server immediately. Connected clients are not notified.

Accepted abbreviation: `ha ca`.

## Usage

Use this command from the command line only; if used in the A3270 server’s initialization file, an error will occur.

## Example

```
halt
The A3270 server will shut down when all the clients have
closed their connections.
The A3270 server is now exiting.
Ready;
```

# help

## Purpose

Use this command to request help information for A3270 server commands.

## Format



## Parameters

### *command*

An A3270 server command. Use this parameter to request help for a specific A3270 server command.

Accepted abbreviation *he*

## Usage

Use this command from the command line only; if used in the A3270 server's initialization file, an error will occur.

## Example

- This example displays the topics for which help is available:

```
help
```

The following commands are available from the console:

```
Display <options> - Display information
TErminate <client> - End a client session
HElp <command>   - Display help for a command
HAIt             - Stop the server
SELftest        - Perform server self-tests
SET <options>    - Change server settings
```

For detailed information on any command, type help <command>. For example, help selftest gives help on the selftest command.

- This example displays help for the display command:

```
help display
```

The DIspay command is used to display information about the A3270 server. There are several options to the DIspay command, as follows:

```
DIspay CLients - list clients currently connected to
                this server
DIspay POLLing - Display the polling interval
DIspay INActivity TIMEOUT - show the timeout for inactive
                clients
DIspay STATistics - shows statistics gathered by the
                server
```

More details on any of the above commands can be obtained by typing help <command>; for example, help display clients gives more details on the display clients command.

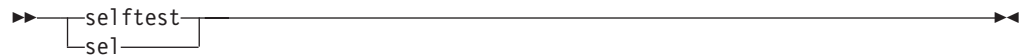
# selftest

## Purpose

Use this command to start the A3270 server internal tests. When the selftest command is issued, the A3270 server performs a series of internal tests to determine whether it is operating correctly. If one or more internal tests fail, one or more error messages is displayed. When an internal test fails, write down the message number or numbers and contact the IBM Software Support Center.

Acceptable abbreviation: sel

## Format



## Parameters

None.

## Usage

Use this command from the command line only; if used in the A3270 server's initialization file, an error will occur.

## Example

- In this example, the A3270 server internal tests completed without errors.

```
sel
Beginning self tests
Self tests have completed.
```

- In this example, the A3270 self tests completed with one error.

**Note:** When the selftest command completes with errors, contact the IBM Software Support Center.

```
sel
Beginning self tests
The incoming queue as lost 1 element.
Self tests have completed.
```

# set

## Purpose

Use this command to set A3270 server options.

## Format

►►—set—| parameters |—inactivity timeout—*n*—polling—*n*—►►

## Parameters

### inactivity timeout *n*

Specifies the amount of time in seconds after which an inactive client is disconnected from the A3270 server. If no data is sent to or received from a client for *n* seconds, the A3270 server ends its connection to the client.

Valid range in seconds: 0 - 9999999.

Default: 0

When the inactivity timeout parameter is set to zero, inactive clients are never disconnected from the server.

Accepted abbreviation: set in ti n

### polling *n*

Sets the frequency in seconds at which the A3270 server polls A3270 clients. Normally, when an A3270 client ends its connection to the A3270 server, the server is notified. However, network conditions can prevent the notification from reaching the server. When polling is set to a non-zero value, the A3270 server polls each client at the interval set on the polling parameter to determine which clients are no longer connected.

Valid range in seconds: 120 - 9999999 and zero. When polling is set to 0, clients are never polled.

Default: 0

Accepted abbreviation: set po

## Usage

Use this command from either the command line or in the A3270 server's initialization file. In the initialization file, multiple set commands are allowed. The set command parameter value specified in the initialization file can be changed temporarily by issuing a set command from the command line. The parameter value specified on the command line is used until the set command is issued again or the A3270 server is restarted.

If the set command is not specified in the initialization file, the parameters are set to defaults when the A3270 server is started.

## Example

- This example sets the inactivity timeout parameter to zero:  
set in ti 0  
Inactive clients will not be terminated.
- This example sets the inactivity timeout parameter to 600 seconds (10 minutes):  
set in ti 600  
Inactive clients will be terminated after 600 seconds:
- This example sets the polling parameter to 600 seconds (10 minutes):



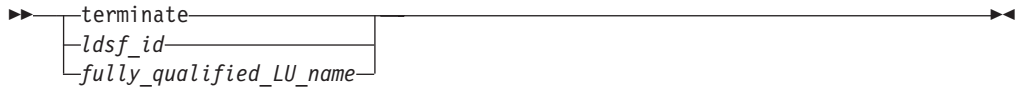
```
set po 600  
Clients will be polled every 600 seconds.
```

# terminate

## Purpose

Use this command to end a connection to an A3270 client.

### VM Format:



### MVS Format:



## Parameters

### ldsfi\_id (VM only)

The Logical Device Support Facility ID (LDSF ID) used for the connection between this A3270 client and the A3270 server. To determine the LDSF ID of the client you want to terminate, either enter the A3270 display command to display all active clients, or you can enter the CMS query command to query the client's user ID and LDSF ID.

### local\_lu\_name (MVS only)

The LU name of the LU 2 used for the connection between this A3270 client and the A3270 server. To determine the LU name of the client you want to terminate, either enter the A3270 display command to display all active clients, or enter the CMS query command to query the client's user ID and LU name. To determine the LU name of the client you want to terminate, use the A3270 display command to display all active clients.

### fully\_qualified\_lu\_name

On VM, specifies the fully qualified LU name of the system on which this client is running. When you use the fully qualified LU parameter, all connections from the clients's system to the A3270 server are ended.

On MVS, specifies the ACBNAME of the LU 2 used for the connection between this A3270 client and the A3270 server.

**VM Examples:** For these examples, assume the A3270 display clients command was used to display the following list of connected clients:

LDSF ID	Partner LU Name	Read path	Write path	State
L009A	NETID.LUA	2	1	Active
L0004	NETID.LUT	4	3	Active
L00D7	NETID.LUT	6	5	Active
L004B	NETID.LUF	8	7	Active
L00C9	NETID.LUF	A	9	Active
L00A0	NETID.LUP	1A	B	Active

- Terminate the connection to the client with the LDSFID of L009A. In this example, one connection is terminated.

```
terminate l009a
APP1023I 1 connection to L009A is being terminated.
```

- Terminate the connections to the client with the fully qualified LU name of NETID.LUT. In this example, two connections are terminated.

```
terminate netid.lut
APP1024I 2 connections to NETID.LUT are being terminated.
```

**MVS Examples:** For these examples, assume the A3270 display clients command was used to display the following list of connected clients:

LU Name	Partner LU Name	Read path	Write path	State
A32T0003	NETID.LUA	2	1	Active
A32T0004	NETID.LUT	4	3	Active
A32T0005	NETID.LUT	6	5	Active
A32T0007	NETID.LUF	8	7	Active
A32T0008	NETID.LUF	A	9	Active
A32T0009	NETID.LUP	1A	B	Active

- Terminate the connection to the client with the LU Name of A32T0003. In this example, one connection is terminated.

```
terminate A32T0003
1 connection to A32T0003 is being terminated
```

- Terminate the connections to the client with the partner LU name of NETID.LUT. In this example, two connections are terminated.

```
terminate netid.lut
2 connections to NETID.LUT are being terminated.
```



---

## Appendix A. How to Read a Syntax Diagram

The syntax diagram shows you how to specify a command so that the operating system can correctly interpret what you type. Read the syntax diagram from left to right and from top to bottom, following the horizontal line (the main path).

---

### Symbols and Punctuation

The following symbols are used in syntax diagrams:

- ▶▶ Marks the beginning of the command syntax.
- ▶ Indicates that the command syntax is continued.
- | Marks the beginning and end of a fragment or part of the command syntax.
- ◀◀ Marks the end of the command syntax.

You must include all punctuation such as colons, semicolons, commas, quotation marks, and minus signs that are shown in the syntax diagram.

---

### Parameters

The following types of parameters are used in syntax diagrams.

#### Required

Required parameters are displayed on the main path.

#### Optional

Optional parameters are displayed below the main path.

#### Default

Default parameters are displayed above the main path.

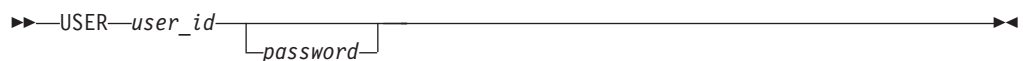
Parameters are classified as keywords or variables. Keywords are displayed in uppercase letters and can be entered in uppercase or lowercase. For example, a command name is a keyword.

Variables are italicized, appear in lowercase letters, and represent names or values you supply. For example, a data set is a variable.

---

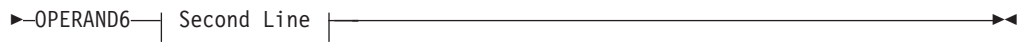
### Syntax Examples

In the following example, the `USER` command is a keyword. The required variable parameter is `user_id`, and the optional variable parameter is `password`. Replace the variable parameters with your own values.



**Longer than one line:** If a diagram is longer than one line, the first line ends with a single arrowhead and the second line begins with a single arrowhead.



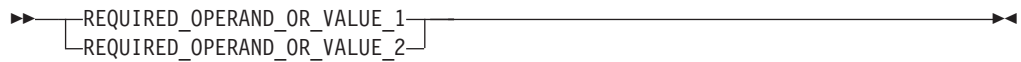


**Required operands:** Required operands and values appear on the main path line.



You must code required operands and values.

**Choose one required item from a stack:** If there is more than one mutually exclusive required operand or value to choose from, they are stacked vertically in alphanumeric order.



**Optional values:** Optional operands and values appear below the main path line.



You can choose not to code optional operands and values.

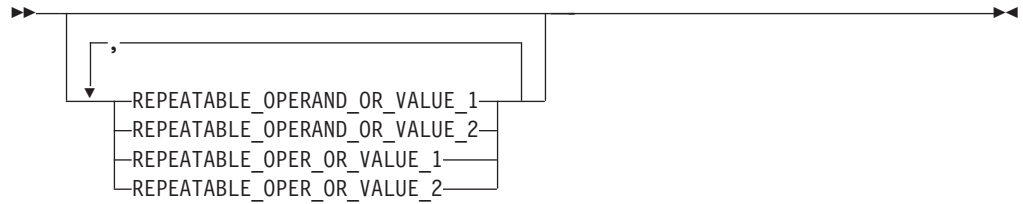
**Choose one optional operand from a stack:** If there is more than one mutually exclusive optional operand or value to choose from, they are stacked vertically in alphanumeric order below the main path line.



**Repeating an operand:** An arrow returning to the left above an operand or value on the main path line means that the operand or value can be repeated. The command means that each operand or value must be separated from the next by a comma.



**Selecting more than one operand:** An arrow returning to the left above a group of operands or values means more than one can be selected, or a single one can be repeated.



If an operand or value can be abbreviated, the abbreviation is described in the text associated with the syntax diagram.

**Case Sensitivity:** VTAM commands are not case sensitive. You can code them in uppercase or lowercase.

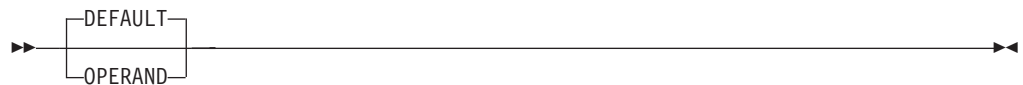
**Nonalphanumeric characters:** If a diagram shows a character that is not alphanumeric (such as parentheses, periods, commas, and equal signs), you must code the character as part of the syntax. In this example, you must code OPERAND=(001,0.001).



**Blank spaces in syntax diagrams:** If a diagram shows a blank space, you must code the blank space as part of the syntax. In this example, you must code OPERAND=(001 FIXED).



**Default operands:** Default operands and values appear above the main path line. VTAM uses the default if you omit the operand entirely.



**Variables:** A word in all lowercase italics is a *variable*. Where you see a variable in the syntax, you must replace it with one of its allowable names or values, as defined in the text.



**Syntax fragments:** Some diagrams contain syntax fragments, which serve to break up diagrams that are too long, too complex, or too repetitious. Syntax fragment names are in mixed case and are shown in the diagram and in the heading of the fragment. The fragment is placed below the main diagram.

▶▶ | Reference to Syntax Fragment | ◀◀

**Syntax Fragment:**

|—1ST\_OPERAND,2ND\_OPERAND,3RD\_OPERAND—|

References to syntax notes appear as numbers enclosed in parentheses above the line. Do not code the parentheses or the number.

(1)  
▶▶—OPERAND—◀◀

**Notes:**

- 1 An example of a syntax note.



---

## Appendix B. A3270 Default Logmodes

The A3270 server supports the VTAM default logmodes.

Table 6 lists the supported device names supported by the A3270 server and the corresponding VTAM default logmodes.

For information about the default logmodes, see the *VTAM Resource Definition Reference*

Table 6. Device Name and Default Logmode Table for A3270

Device Name	Default Logmode
3277	D4B32782
3270-2-E	NSX32702
3278-2	D4B32782
3278-3-E	NSX32702
3278-3	D4B32783
3278-4-E	NSX32702
3278-4	D4B32784
3278-5-E	NSX32702
3278-5	D4B32785
3279-2-E	NSX32702
3279-2	D4B32782
3279-3-E	NSX32702
3279-3	D4B32783
2379-4-E	NSX32702
3270-4	D4B32784
3279-5-E	NSX32702
3279-5	D4B32785



---

## Appendix C. Configuring APPC to Support APPC Security SAME

This section describes how APPC security=SAME can be helpful to users in an APPC network, and how APPC/VM and APPC/MVS must be configured to make use of this facility. There are certain network configuration considerations for using security=SAME for APPC conversations. This does not require changing the configuration of the APPC Application Suite programs, but rather the configuration of the underlying APPC subsystems, such as APPC/VM, APPC/MVS, and Communications Manager/2.

---

### What is APPC security=SAME?

APPC security=SAME is used to provide conversation security without the user having to explicitly enter a user ID and a password. This means that the user identifies himself or herself to the partner computer.

### How is security=SAME Helpful to Users?

The security type SAME can be a great convenience to users of the APPC Application Suite. Ways that users can take advantage of security=SAME include:

- ANAME administrators may wish to use security=SAME if the administrator has a user ID on the system where the ANAME server is installed. The administrator would then use the -c flag on the ANAME command to identify himself or herself as an administrator.
- A user wishing to test connectivity between two systems on which he or she has an account may use the APING program with security=SAME to avoid entering a user ID and password.
- Individuals transferring files from workstations to VM or MVS hosts may use security=SAME if their IDs and passwords match between the workstation and the host. Since security=SAME is the default for the ACOPY program, the user would not need to remember to enter any flags for security purposes.

Another advantage of security=SAME over security=PROGRAM is that security=PROGRAM sends a user ID and password over the network *in the clear*. In other words, the user ID and password are not encrypted. When using security=SAME, APPC does not always transmit the password. For more details, including whether a user ID and password is transmitted over the network, see "How is APPC security=SAME Implemented?".

### How is APPC security=SAME Implemented?

In order to understand what must be configured to use security=SAME, it is helpful to understand how security=SAME is implemented. This section describes what is done by the underlying APPC subsystem when an application requests security=SAME.

When security=SAME is requested by an application, one of the following mechanisms is used for the conversation. The mechanisms are tried in the order listed below:

1. An already-verified user ID
2. Persistent verification of user IDs
3. Enhanced security SAME
4. Downgrade to security=NONE

## **Already-verified user IDs**

When an APPC program requests security=SAME, the underlying APPC subsystem attempts to use already-verified user IDs (already-verified) if the server's LU accepts it. When already-verified is used, the LU that initiates the conversation sends a user ID and an indication that the password for that user ID has already been checked.

***Already-verified in APPC/MVS:*** APPC/MVS supports accepting the already-verified indicator on inbound conversations. If APPC/MVS is configured to accept the already-verified indicator, any user ID which is received with the already-verified indicator will be accepted. APPC/MVS will also send the already-verified indicator to a partner LU on an outbound conversation if that LU accepts the already-verified indicator.

***Already-verified in APPC/VM:*** APPC/VM supports accepting the already-verified indicator on inbound conversations. User IDs received by APPC/VM with the already-verified indicator must be listed in an AGW ADD USERID statement in AVS. If the user ID does not match one of the user IDs defined to AVS on an AGW ADD USERID statement, the conversation is rejected, and the client receives a conversation security failure (indicated by a message such as APP0207E). For details about the AGW ADD USERID statement, see *VM/ESA Connectivity Planning, Administration, and Operation*

APPC/VM will send the already-verified indicator to a partner LU on an outbound conversation if that LU accepts the already-verified indicator.

## **Persistent Verification of user IDs**

Persistent verification of user IDs (persistent verification) provides a mechanism where a password is only sent on the first conversation for any given user ID between two LUs. When persistent verification is used, the first time a user requests a conversation between two LUs using security=SAME, the user's user ID and password are extracted from the client's environment and sent to the server system. The user ID and password are verified, and the conversation proceeds. For all subsequent conversations between those LUs, if the same user requests a conversation using security=SAME, the user ID is sent along with an indicator that the password has previously been verified by the server's LU.

***Persistent verification of user IDs in APPC/MVS:*** APPC/MVS supports accepting persistent verification on inbound conversations. Once the user ID and password are verified by the local security manager, the conversation is accepted. APPC/MVS does not support sending persistent verification on outbound conversations.

***Persistent verification of user IDs in APPC/VM:*** APPC/VM does not support persistent verification for either inbound or outbound conversations.

## **Enhanced security SAME**

Enhanced security SAME is used when the system initiating the conversation supports this facility and has a user ID and password available to it.

The enhanced security SAME option will send the local user ID and password, exactly as if they had been entered by the user (this appears to the partner as if security=PROGRAM were being used). This means that the user ID and password that the user logs on to the local system with must match that user's user ID and password on the partner system. In other words, if Mark has a user ID of MEPETERS and a password of ABCXYZ on system MVS1, he must log on to his local system with a user ID of MEPETERS and a password of ABCXYZ.

**Enhanced security SAME on APPC/MVS:** On conversations inbound into APPC/MVS, enhanced security SAME appears as if the partner were using security=PROGRAM. Therefore, APPC/MVS supports inbound conversations using enhanced security SAME. APPC/MVS does not support using enhanced security SAME for outbound conversations.

**Enhanced security SAME on APPC/VM:** On conversations inbound into APPC/VM, enhanced security SAME appears as if the partner were using security=PROGRAM. Therefore, APPC/VM supports inbound conversations using enhanced security SAME. APPC/VM does not support using enhanced security SAME for outbound conversations.

### **Downgrade to security=NONE**

All APPC subsystems support using security=NONE. Security=SAME will be downgraded to security=NONE automatically when none of the mechanisms listed above are available. This can happen because the mechanisms are not supported by one of the LUs involved, or because one of the LUs involved is configured not to accept the security mechanisms that it supports.

When a conversation is downgraded to security=NONE, it appears to the server's LU that the client had requested security=NONE. The client is not informed when security=SAME is downgraded to security=NONE. However, programs such as AFTP, which support determining the user ID for a conversation, will be able to detect that security=NONE is being used.

**Downgrading to security=NONE in APPC/MVS:** APPC/MVS supports downgrading security=SAME to security=NONE on both inbound and outbound conversations.

**Downgrading to security=NONE in APPC/VM:** APPC/VM supports downgrading security=SAME to security=NONE on both inbound and outbound conversations.

---

## **Network Configuration**

This section describes the configuration parameters which can be set for APPC/VM and APPC/MVS, and provides suggestions for configuring APPC on each of these platforms.

### **APPC/MVS Parameters**

The type of security accepted by APPC/MVS is defined by the application program major node definition in VTAM for the APPC LU given in the APPCPMxx member. The APPC LU is defined in application program major node definition for APPC/MVS.

One optional parameter on an application program major node definition is the SECACPT= parameter. The value of this parameter determines what security types the LU will accept. A list of possible values can be found in the *z/OS Communications Server: SNA Resource Definition Reference*. That list is summarized below:

#### **ALREADYV**

Already-verified user IDs, enhanced security SAME, downgrading to security=NONE.

**AVPV** Already-verified user IDs, persistent verification of user IDs, enhanced security SAME, downgrading to security=NONE.

**PERSISTV**

Persistent verification of user IDs, downgrading to security=NONE.

**CONV** Enhanced security SAME, downgrading to security=NONE.

**NONE** Downgrading to security=NONE.

The SECACPT= parameter does not affect the security mechanisms that will be used for outbound conversations. APPC/MVS will always attempt to use already-verified user IDs on outbound conversations, downgrading to security=NONE if already-verified is not accepted by the partner LU.

## Parameters for APPC/VM

The type of security that will be accepted by APPC/VM is defined in the application program major node definitions in VTAM for the global and private AVS gateways.

One optional parameter on an application program major node definition is the SECACPT= parameter. The value of this parameter determines what security types the LU will accept. A list of possible values is given in the *VTAM Resource Definition Reference*. That list is summarized below:

**ALREADYV**

Already-verified user IDs, enhanced security SAME, downgrading to security=NONE.

**CONV** Enhanced security SAME, downgrading to security=NONE.

**NONE** Downgrading to security=NONE.

The SECACPT= parameter does not affect the security mechanisms that will be used for outbound conversations. APPC/VM will always attempt to use already-verified user IDs on outbound conversations, downgrading to security=NONE if already-verified is not accepted by the partner LU.

## Recommendations for APPC/MVS

If an APPC/MVS LU accepts the already-verified indicator, the LU should only accept conversations from other LUs which can be trusted. User IDs coming in with the already-verified indicator do not have a password associated with them; the partner system is assumed to have already verified the password. Therefore, if the already-verified indicator is accepted from LUs which cannot be trusted, a possible security exposure exists. The configuration suggested below will avoid this problem.

For the APPC Application Suite, the following configuration is recommended:

- Define one APPC/MVS LU which accepts the already-verified indicator (code SECACPT=ALREADYV on the application program major node definition). Set up this LU so that it requires LU-LU security. Define the passwords that will be used for LU-LU security, and give those passwords only to systems which can be trusted to verify the passwords for any user IDs they send, such as VM and MVS.

See *VTAM Resource Definition Reference* for details about LU-LU security.

- Define a second APPC/MVS LU which does not accept the already-verified indicator (code SECACPT=PERSISTV on the application program major node definition). Set up this LU so that it does not require LU-LU security. This LU should be used by users who are located on non-trusted systems, such as desktop workstations.

## Recommendations for APPC/VM

For the APPC Application Suite, the following configuration is recommended:

- Specify SECACPT=ALREADYV on the application program major node definitions for the AVS gateways (two gateways, a global gateway and a private gateway, should be defined in AVS).
- Enter AGW ADD USERID statements for any users on VM or MVS hosts wishing to use security=SAME to contact this system.

The AGW ADD USERID statements can be placed in the PROFILE GCS file for the AVS virtual machine, so they will be automatically entered every time AVS is restarted. The *luname* parameter on the AGW ADD USERID statement can be used to accept the already-verified indicator on user IDs only from LUs which are known to have verified the password for that user ID.





## Appendix D. Information Apars

This appendix lists information apars for IP and SNA books.

### Notes:

1. Information apars contain updates to previous editions of the manuals listed below. Books updated for V1R2 are complete except for the updates contained in the information apars that may be issued after V1R2 books went to press.
2. Information apars are predefined for z/OS V1R2 Communications Server and may not contain updates.

## IP Information Apars

Table 7 lists information apars for IP books.

Table 7. IP Information Apars

Title	z/OS CS V1R2	CS for OS/390 2.10 and z/OS CS V1R1	CS for OS/390 2.8	CS for OS/990 2.7	CS for OS/390 2.6	CS for OS/390 2.5
IP API Guide	ii12861	ii12371	ii11635	ii11558	ii11405	ii11144
IP CICS Sockets Guide	ii12862		ii11626	ii11559	ii11406	ii11145
IP Configuration			ii11620 ii12068 ii12353 ii12649	ii11555 ii11637 ii11995 ii12325	ii11402 ii11619 ii12066 ii12455	ii11159 ii11979 ii12315
IP Configuration Guide	ii12498	ii12362 ii12493				
IP Configuration Reference	ii12499	ii12363 ii12494 ii12712				
IP Diagnosis	ii12503	ii12366 ii12495	ii11628	ii11565	ii11411	ii11160 ii11414
IP Messages Volume 1	ii12857	ii12367	ii11630	ii11562	ii11408	ii11636
IP Messages Volume 2	ii12858	ii12368	ii11631	ii11563	ii11409	ii11281
IP Messages Volume 3	ii12859	ii12369	ii11632 ii12883	ii11564 ii12884	ii11410 ii12885	ii11158
IP Messages Volume 4	ii12860					
IP Migration	ii12497	ii12361	ii11618	ii11554	ii11401	ii11204
IP Network Print Facility	ii12864		ii11627	ii11561	ii11407	ii11150
IP Programmer's Reference	ii12505		ii11634	ii11557	ii11404	ii12496

Table 7. IP Information Apars (continued)

Title	z/OS CS V1R2	CS for OS/390 2.10 and z/OS CS V1R1	CS for OS/390 2.8	CS for OS/990 2.7	CS for OS/390 2.6	CS for OS/390 2.5
IP and SNA Codes	ii12504	ii12370	ii11917	Added TCP/IP codes to VTAM codes V2R6 ii11611	ii11361	ii11146 ii11097
IP User's Guide		ii12365	ii11625	ii11556	ii11403	ii11143
IP User's Guide and Commands	ii12501					
IP System Admin Guide	ii12502					
Quick Reference	ii12500	ii12364				

## SNA Information Apars

Table 8 lists information apars for SNA books.

Table 8. SNA Information Apars

Title	z/OS CS V1R2	CS for OS/390 2.10 and z/OS CS V1R1	CS for OS/390 2.8	CS for OS/390 2.7	CS for OS/390 2.6	CS for OS/390 2.5
Anynet SNA over TCP/IP			ii11922	ii11633	ii11624	ii11623
Anynet Sockets over SNA			ii11921	ii11622	ii11519	ii11518
CSM Guide						
IP and SNA Codes		ii12370	ii11917	ii11611	ii11361	ii11097
SNA Customization	ii12872	ii12388	ii11923	ii11925 ii12008	ii11924 ii12007	ii11092 ii11621 ii12006
SNA Diagnosis	ii12490	ii12389	ii11915	ii11615	ii11357	ii11585
SNA Messages	ii12491	ii12382	ii11916	ii11610	ii11358	ii11096
SNA Network Implementation Guide	ii12487	ii12381	ii11911	ii11609 ii12683	ii11353 ii11493	ii11095
SNA Operation	ii12489	ii12384	ii11914	ii11612	ii11355	ii11098
SNA Migration	ii12486	ii12386	ii11910	ii11614	ii11359	ii11100
SNA Programming		ii12385	ii11920	ii11613	ii11360	ii11099
Quick Reference	ii12500	ii12364	ii11913	ii11616	ii11356	
SNA Resource Definition Reference	ii12488	ii12380 ii12567	ii11912 ii12568	ii11608 ii12569	ii11354 ii12259 ii12570	ii11094 ii11151 ii12260 ii12571
SNA Resource Definition Samples						

---

## Appendix E. Notices

IBM may not offer all of the products, services, or features discussed in this document. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:**

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs

and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Site Counsel  
IBM Corporation  
P.O.Box 12195  
3039 Cornwallis Road  
Research Triangle Park, North Carolina 27709-2195  
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly

tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. \_enter the year or years\_. All rights reserved.

This product includes cryptographic software written by Eric Young.

If you are viewing this information softcopy, photographs and color illustrations may not appear.

You can obtain softcopy from the z/OS Collection (SK3T-4269), which contains BookManager and PDF formats of unlicensed books and the z/OS Licensed Product Library (LK3T-4307), which contains BookManager and PDF formats of licensed books.

---

## Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM	Micro Channel
Advanced Peer-to-Peer Networking	MVS
AFP	MVS/DFP
AD/Cycle	MVS/ESA
AIX	MVS/SP
AIX/ESA	MVS/XA
AnyNet	MQ
APL2	Natural
APPN	NetView
AS/400	Network Station
AT	Nways
BookManager	Notes
BookMaster	NTune
CBPDO	NTuneNCP
C/370	OfficeVision/MVS
CICS	OfficeVision/VM
CICS/ESA	Open Class
C/MVS	OpenEdition
Common User Access	OS/2
C Set ++	OS/390
CT	OS/400
CUA	Parallel Sysplex
DATABASE 2	Personal System/2
DatagLANce	PR/SM
DB2	PROFS
DFSMS	PS/2
DFSMSsdfp	RACF
DFSMSshsm	Resource Link
DFSMS/MVS	Resource Measurement Facility
DPI	RETAIN
Domino	RFM
DRDA	RISC System/6000
eNetwork	RMF
Enterprise Systems Architecture/370	RS/6000
ESA/390	S/370
ESCON	S/390
@server	SAA
ES/3090	SecureWay
ES/9000	Slate
ES/9370	SP
EtherStreamer	SP2
Extended Services	SQL/DS
FAA	System/360

FFST	System/370
FFST/2	System/390
FFST/MVS	SystemView
First Failure Support Technology	Tivoli
GDDM	TURBOWAYS
Hardware Configuration Definition	UNIX System Services
IBM	Virtual Machine/Extended Architecture
IBMLink	VM/ESA
IBMLINK	VM/XA
IMS	VSE/ESA
IMS/ESA	VTAM
InfoPrint	WebSphere
Language Environment	XT
LANStreamer	z/Architecture
Library Reader	z/OS
LPDA	zSeries
MCS	400
	3090
	3890

Lotus, Freelance, and Word Pro are trademarks of Lotus Development Corporation in the United States, or other countries, or both.

Tivoli and NetView are trademarks of Tivoli Systems Inc. in the United States, or other countries, or both.

DB2 and NetView are registered trademarks of International Business Machines Corporation or Tivoli Systems Inc. in the U.S., other countries, or both.

The following terms are trademarks of other companies:

ATM is a trademark of Adobe Systems, Incorporated.

BSC is a trademark of BusiSoft Corporation.

CSA is a trademark of Canadian Standards Association.

DCE is a trademark of The Open Software Foundation.

HYPERchannel is a trademark of Network Systems Corporation.

UNIX is a registered trademark in the United States, other countries, or both and is licensed exclusively through X/Open Company Limited.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ActionMedia, LANDesk, MMX, Pentium, and ProShare are trademarks of Intel Corporation in the United States, other countries, or both. For a complete list of Intel trademarks, see <http://www.intel.com/tradmarx.htm>.

Other company, product, and service names may be trademarks or service marks of others.





---

# Index

## A

- a3270 command 68
- A3270 commands
  - a3270 68
  - display 69
  - halt 71
  - help 72
  - selftest 73
  - set 74
  - terminate 76
- A3270 server
  - ACBNAME 29
  - activating the application program major node 29
  - APP3APPE utility 32
  - APP3INME utility 45
  - APPC APPL definition statement 29
  - ATCCON 33
  - automatic start-up for the A3270 server 33
  - client emulators 63
  - commands 66
  - controlling access to specific applications 42
  - create a logon profile 12
  - creating the A3270 initialization file, MVS 45
  - creating the VTAM application program major node 32
  - default application 42
  - defining LU 2s to the A3270 server 42
  - defining the default application 42
  - defining the server to VTAM, MVS 29
  - description 2
  - filenames, server (VM) 10
  - list of VM configuration tasks 10
  - locating server files on VM 10
  - LU type 2 APPL definition statement 29
  - managing 63
  - MAXCLIENTS 42, 43
  - modifying the server REXX EXEC, MVS 33
  - MVS configuration tasks 29
  - overview 63
  - privilege class 11
  - RH and sequence number support 63
  - sample initialization file, MVS 43
  - sample initialization file, VM 42
  - sample major node definition 29
  - sample VTAM application program major node definition 31
  - server's virtual machine 11
  - setting up the initialization file, MVS 42
  - setting up the initialization file, VM 41
  - specifying the maximum number of clients, MVS 42
  - specifying the maximum number of clients, VM 41
  - starting 65, 68
  - starting the application program major node 33
  - stopping 65, 71
  - synchronization 63
  - terminal type support 63
  - VBUILD definition statement 29
  - A3270 server (*continued*)
    - virtual storage 11
- ACBNAME 29
- ACOPY
  - modifying the REXX EXEC, MVS 16
  - MVS sample 16
  - updating ACOPY EXEC on MVS 16
- add duplicate user names to the ANAME database 54
- AFTP
  - adding a transaction program definition (VM) 4
  - adding side information (MVS) 18
  - adding side information (VM) 4
  - description 2
  - EXEC, updating 15
  - filenames (VM) 3
  - list of MVS configuration tasks 15
  - list of VM configuration tasks 3
  - locating files on VM 3
  - managing AFTP servers 47
  - modify the AFTP server REXX EXEC, MVS 16
  - modifying the AFTP TP profile 20
  - sample initialization file 35, 37
  - setting up the initialization file 35
  - TPMODIFY 20
  - updating the logon profile 5
- AFTP EXEC, MVS
  - sample 15
  - updating the exec 15
- AFTPD audit log 47
- ANAME
  - adding side information, MVS 25
  - adding side information (VM) 7
  - create a logon profile 10
  - database
    - add duplicate user names 54
    - administration tasks 52
    - backup the database file 55
    - delete records for other users 54
    - recover the database file 55
    - register records for other users 54
  - description 2
  - filenames, server (VM) 8
  - filenames (VM) 7
  - list of VM configuration tasks, client 7
  - list of VM configuration tasks, server 8
  - locating files on VM 7
  - locating server files on VM 8
  - modifying the client REXX EXEC, MVS 24
  - MVS configuration tasks 24
  - privilege class 8
  - sample initialization file 40
  - sample side information JCL 26
  - server's virtual machine 8
  - setting up the initialization file 38
  - SIADD 26
  - SIMODIFY 26
  - starting the ANAME server 51, 61
  - stopping the ANAME server 51, 62

- ANAME (*continued*)
  - system administrator
    - defining 38, 53
    - set trace 40
  - TPADD 29
  - TPMODIFY 29
  - virtual storage 8
- ANAME commands
  - aname register 59
  - anamed 61
  - delete 57
  - halt 8, 62
- ANAME database
  - add duplicate user names 54
  - administration tasks 52
  - backup the database file 55
  - delete records for other users 54
  - recover the database file 55
  - register records for other users 54
- anamed command 61
- APING
  - adding side information, MVS 21
  - adding side information (VM) 6
    - description 1
    - filenames (VM) 5
    - list of VM configuration tasks 5
    - locating files on VM 5
    - modifying the REXX EXEC 20
    - MVS configuration tasks 20
    - side information JCL 22
    - SIMODIFY 22
    - updating the logon profile 7
- APING server
  - adding a transaction program profile, MVS 22, 26
  - modifying the ANAMED TP profile, MVS 29
  - modifying the APINGD TP profile 24
    - TPADD 24
    - TPMODIFY 24
- APINGD
  - adding a transaction program profile, MVS 22, 26
  - modifying the ANAMED TP profile, MVS 29
  - modifying the APINGD TP profile 24
    - TPADD 24
    - TPMODIFY 24
- APP3A3MP 33
- APP3APPE utility 32
- APP3INME utility 45
- APPC APPL definition statement 29
- APPC Application Suite
  - configuring on MVS 13
  - configuring on VM 3
  - installation (MVS) 13
  - installation (VM) 3
  - messages, installing 13
- APPC Emulation Server
  - ACBNAME 29
  - activating the application program major node 29
  - APP3APPE utility 32
  - APP3INME utility 45
  - APPC APPL definition statement 29
  - ATCCON 33
- APPC Emulation Server (*continued*)
  - automatic start-up for the A3270 server 33
  - client emulators 63
  - commands 66
  - controlling access to specific applications 42
  - create a logon profile 12
  - creating the A3270 initialization file, MVS 45
  - creating the VTAM application program major node 32
    - default application 42
    - defining LU 2s to the A3270 server 42
    - defining the default application 42
    - defining the server to VTAM, MVS 29
    - description 2
    - filenames, server (VM) 10
    - list of VM configuration tasks 10
    - locating server files on VM 10
    - LU type 2 APPL definition statement 29
    - managing 63
    - MAXCLIENTS 42, 43
    - modifying the server REXX EXEC, MVS 33
    - MVS configuration tasks 29
    - overview 63
    - privilege class 11
    - RH and sequence number support 63
    - sample initialization file, MVS 43
    - sample initialization file, VM 42
    - sample major node definition 29
    - sample VTAM application program major node definition 31
    - server's virtual machine 11
    - setting up the initialization file, MVS 42
    - setting up the initialization file, VM 41
    - specifying the maximum number of clients, MVS 42
    - specifying the maximum number of clients, VM 41
    - starting 65, 68
    - starting the application program major node 33
    - stopping 65, 71
    - synchronization 63
    - terminal type support 63
    - VBUILD definition statement 29
    - virtual storage 11
- APPC File Transfer Protocol
  - adding a transaction program definition (VM) 4
  - adding side information (MVS) 18
  - adding side information (VM) 4
    - description 2
    - EXEC, updating 15
    - filenames (VM) 3
    - list of MVS configuration tasks 15
    - list of VM configuration tasks 3
    - locating files on VM 3
    - managing AFTP servers 47
    - modify the AFTP server REXX EXEC, MVS 16
    - modifying the AFTP TP profile 20
    - sample initialization file 35, 37
    - setting up the initialization file 35
    - TPMODIFY 20
    - updating the logon profile 5
- APPC NameServer
  - adding side information, MVS 25

APPC NameServer (*continued*)

- adding side information (VM) 7
- create a logon profile 10
- database
  - add duplicate user names 54
  - administration tasks 52
  - backup the database file 55
  - delete records for other users 54
  - recover the database file 55
  - register records for other users 54
- description 2
- filenames, server (VM) 8
- filenames (VM) 7
- list of VM configuration tasks, client 7
- list of VM configuration tasks, server 8
- locating files on VM 7
- locating server files on VM 8
- modifying the client REXX EXEC, MVS 24
- MVS configuration tasks 24
- privilege class 8
- sample initialization file 40
- sample side information JCL 26
- server's virtual machine 8
- setting up the initialization file 38
- SIADD 26
- SIMODIFY 26
- starting the ANAME server 51, 61
- stopping the ANAME server 51, 62
- system administrator
  - defining 38, 53
  - set trace 40
- TPADD 29
- TPMODIFY 29
- virtual storage 8

APPC security same, configuration 85

APPC/VM VTAM Support (AVS) 3

APPFACMX

- modifying the REXX EXEC, MVS 16
- MVS sample 16
- updating ACOPY EXEC on MVS 16

APPFAMX 15

APPFSIAJ 18

application program major node 29

APPMANMX 24

APPMSIAJ 25

APPPAPMX 20

APPPSIAJ 21

ATCCON 33

audit log, AFTPD 47

AVS (APPC/VM VTAM Support) 3

## B

backup the ANAME database 55

## C

command reference, A3270 66

commands, A3270

- a3270 68
- display 69

commands, A3270 (*continued*)

- halt 71
- help 72
- selftest 73
- set 74
- terminate 76

commands, ANAME

- aname register 59
- anamed 61
- delete 57
- halt 8, 62

commands used to

- start the A3270 server 68
- stop the A3270 server 71

configuration tasks, MVS

- A3270 default application 42
- A3270 server 29
- adding a transaction program profile 22, 26
- adding ANAME side information 25
- adding APING side information 21
- adding side information, AFTP 18
- AFTP clients and servers 15
- ANAME clients 24
- ANAME servers 26
- APING clients and servers 20
- configuring VTAM to start the A3270 server 33
- creating the A3270 application program major node 32
- creating the A3270 initialization file 45
- defining the A3270 server to VTAM 29
- defining the APPC Application Suite 13
- modify the A3270 server REXX EXEC 33
- modify the ACOPY client REXX EXEC 16
- modify the AFTP client REXX EXEC 15
- modify the AFTP server REXX EXEC 16
- modify the ANAME client REXX EXEC 24
- modify the APING client REXX EXEC 20
- modifying ANAME side information 26
- modifying APING side information 22
- modifying the AFTPD TP profile 20
- modifying the ANAMED TP profile 29
- modifying the APINGD TP profile 24
- prerequisites 13
- sample JCL 13
- setting up the A3270 initialization file, MVS 42
- SIADD 22, 26
- SIMODIFY 26
- starting the A3270 application program major node 33
- system software requirements 13
- TPADD 24, 29
- TPMODIFY 20, 24
- updating the ACOPY EXEC 16
- updating the AFTP EXEC 15

configuration tasks, VM

- A3270 server 10
- adding a transaction program definition, AFTPD 4
- adding side information, AFTP 4
- adding side information, ANAME 7
- adding side information, APING 6
- AFTP clients and servers 3

- configuration tasks, VM *(continued)*
  - ANAME clients 7
  - ANAME server 8
  - APING clients and servers 5
  - create a logon profile, A3270 server 12
  - create a logon profile, ANAME server 10
  - creating CP directory for A3270 server 11
  - creating CP directory for ANAME server 9
  - defining the A3270 server's virtual machine
    - creating a logon profile 12
    - creating CP directory 11
  - defining the ANAME server's virtual machine
    - creating a logon profile 10
    - creating CP directory 9
  - defining the APPC Application Suite 3
  - estimating VM storage requirements for ANAME database 9
  - locating the A3270 server files 10
  - locating the AFTP files 3
  - locating the ANAME client REXX EXEC 7
  - locating the ANAME server files 8
  - locating the APING files 5
  - prerequisites 3
  - setting up the A3270 initialization file, VM 41
  - setting up the AFTP initialization file 35
  - setting up the ANAME initialization file 38
  - system software requirements 3
  - updating the logon profile, AFTPD 5
  - updating the logon profile, APINGD 7
- CP directory, creating A3270 server's 11
- CP directory, creating ANAME server's 9
- CPI-C blank symbolic destination name 21

## D

- database, ANAME
  - add duplicate user names 54
  - administration tasks 52
  - backup the database file 55
  - delete records for other users 54
  - recover the database file 55
  - register records for other users 54
- database administration, ANAME 52
- default application 42
- define ANAME system administrator 38
- defining the A3270 server to VTAM 29
- delete command, ANAME 57
- display command, A3270 69

## F

- file transfer defaults, AFTP 35

## H

- halt, ANAME command 8
- halt command, A3270 71
- halt command, ANAME 62
- help command, A3270 72

## I

- initialization file, A3270 41
- initialization files
  - A3270
    - MVS 42
    - sample, MVS 43
    - sample, VM 42
    - VM 41
  - AFTP
    - examples 36
    - map verb 35
    - options 36
    - sample 35, 37
    - source\_mask 36
    - syntax 35
    - target\_mask 36
  - ANAME
    - sample 40
    - set\_trace 40
    - syntax 38
- installing APPC Application Suite messages 13

## J

- JCL samples
  - A3270 initialization file 43
  - A3270 REXX EXEC 33
  - A3270 VTAM application program major node definition 29, 31
  - ACOPY EXEC 16
  - ANAME add transaction program JCL 26
  - ANAME EXEC 24
  - APING EXEC 20
  - APING transaction program profile 22
  - APP3A3MP 33
  - APPFSIAJ 18
  - APPMANMX 24
  - APPMSIAJ 25
  - APPMPAJ 26
  - APPPAPMX 20
  - APPPSIAJ 21
  - APPPTPAJ 22
  - side information JCL for AFTP 18
  - side information JCL for ANAME 25
  - side information JCL for APING 21

## L

- LU type 2 42
- LU type 2 APPL definition statement 29

## M

- managing AFTP servers 47
- managing ANAME database records 54
- managing the A3270 server 63
- MAXCLIENTS 42, 43
- messages, installing APPC Application Suite 13
- modifying the A3270 server REXX EXEC 33
- modifying the ACOPY client REXX EXEC, MVS 16

- modifying the AFTP client REXX EXEC, MVS 15
- modifying the AFTP server REXX EXEC, MVS 16
- modifying the ANAME client REXX EXEC 24
- modifying the APING client REXX EXEC 20
- MVS configuration tasks
  - A3270 default application 42
  - A3270 server 29
  - adding a transaction program profile 22, 26
  - adding ANAME side information 25
  - adding APING side information 21
  - adding side information, AFTP 18
  - AFTP clients and servers 15
  - ANAME clients 24
  - ANAME servers 26
  - APING clients and servers 20
  - configuring VTAM to start the A3270 server 33
  - creating the A3270 application program major node 32
  - creating the A3270 initialization file 45
  - defining the A3270 server to VTAM 29
  - defining the APPC Application Suite 13
  - modify the A3270 server REXX EXEC 33
  - modify the ACOPIY client REXX EXEC 16
  - modify the AFTP client REXX EXEC 15
  - modify the AFTP server REXX EXEC 16
  - modify the ANAME client REXX EXEC 24
  - modify the APING client REXX EXEC 20
  - modifying ANAME side information 26
  - modifying APING side information 22
  - modifying the AFTP TP profile 20
  - modifying the ANAMED TP profile 29
  - modifying the APINGD TP profile 24
  - prerequisites 13
  - sample JCL 13
  - setting up the A3270 initialization file, MVS 42
  - SIADD 22, 26
  - SIMODIFY 26
  - starting the A3270 application program major node 33
  - system software requirements 13
  - TPADD 24, 29
  - TPMODIFY 20, 24
  - updating the ACOPIY EXEC 16
  - updating the AFTP EXEC 15
- MVS samples
  - A3270 initialization file 43
  - A3270 REXX EXEC 33
  - A3270 VTAM application program major node definition 29, 31
  - ACOPY EXEC 16
  - ANAME add transaction program JCL 26
  - ANAME EXEC 24
  - APING EXEC 20
  - APING transaction program profile 22
  - APP3A3MP 33
  - APPFSIAJ 18
  - APPMANMX 24
  - APPMSIAJ 25
  - APPMTPAJ 26
  - APPPAPMX 20
  - APPPSIAJ 21

- MVS samples (*continued*)
  - APPPTPAJ 22
  - side information JCL for AFTP 18
  - side information JCL for ANAME 25
  - side information JCL for APING 21

## N

- NameServer commands
  - aname register 59
  - anamed 61
  - delete 57
  - halt 8, 62
- NameServer database
  - add duplicate user names 54
  - administration tasks 52
  - backup the database file 55
  - delete records for other users 54
  - recover the database file 55
  - register records for other users 54

## O

- options parameter, AFTP 36

## P

- parameters
  - cancel, A3270 halt command 71
  - clients, A3270 display command 69
  - inactivity timeout, A3270 display command 69
    - A3270 set command 74
  - normal, A3270 halt command 71
  - polling, A3270 display command 70
  - quick, A3270 halt command 71
  - statistics, A3270 display command 70
- prevent\_update 39

## R

- recover the ANAME database file 55
- register command, ANAME 59
- RH and sequence number support 63

## S

- sample JCL
  - A3270 initialization file 43
  - A3270 REXX EXEC 33
  - A3270 VTAM application program major node definition 29, 31
  - ACOPY EXEC 16
  - ANAME add transaction program JCL 26
  - ANAME EXEC 24
  - APING EXEC 20
  - APING transaction program profile 22
  - APP3A3MP 33
  - APPFSIAJ 18
  - APPMANMX 24
  - APPMSIAJ 25

- sample JCL (*continued*)
  - APPMTPAJ 26
  - APPPAPMX 20
  - APPPSIAJ 21
  - APPPTPAJ 22
  - side information JCL for AFTP 18
  - side information JCL for ANAME 25
  - side information JCL for APING 21
- samples, MVS
  - A3270 initialization file 43
  - A3270 REXX EXEC 33
  - A3270 VTAM application program major node
    - definition 29, 31
  - ACOPY EXEC 16
  - ANAME add transaction program JCL 26
  - ANAME EXEC 24
  - APING EXEC 20
  - APING transaction program profile 22
  - APP3A3MP 33
  - APPFSIAJ 18
  - APPMANMX 24
  - APPMSIAJ 25
  - APPMTPAJ 26
  - APPPAPMX 20
  - APPPSIAJ 21
  - APPPTPAJ 22
  - side information JCL for AFTP 18
  - side information JCL for ANAME 25
  - side information JCL for APING 21
- samples, VM
  - A3270 initialization file 42
  - AFTP initialization file 35, 37
  - ANAME initialization file 40
  - side information for AFTP 4
  - side information for ANAME 7
  - side information for APING 6
  - transaction program definition, AFTP 4
- security same, configuration 85
- selftest command, A3270 73
- set command, A3270 74
- set\_trace, ANAME 40
- setting up the A3270 initialization file on MVS 42
- SIADD 19, 22
- side information
  - ANAME JCL sample 25
  - APING JCL sample 21
  - for AFTP (MVS) 18
  - for AFTP (VM) 4
  - for ANAME, MVS 25
  - for ANAME (VM) 7
  - for APING, MVS 21
  - for APING (VM) 6
  - modifying APING side information 22
  - modifying for ANAME 26
  - sample for AFTP (MVS) 18
  - sample for AFTP (VM) 4
  - sample for ANAME (VM) 7
  - sample for APING (VM) 6
  - SIADD 22, 26
  - SIMODIFY 22, 26
  - when required 25

- side information (*continued*)
  - when required (MVS) 18
  - when required for APING 21
- SIMODIFY 19, 22, 26
- source\_mask 36
- starting the A3270 server 65, 68
- starting the ANAME server 51, 61
- stopping the A3270 server 65, 71
- stopping the ANAME server 51, 62
- storage requirements, ANAME database (VM) 9
- synchronization, A3270 support of 63
- syntax diagram, reading 79
- system software requirements (MVS) 13
- system software requirements (VM) 3

## T

- target\_mask 36
- terminal types, A3270 support of 63
- terminate command, A3270 76
- TPADD 20, 24, 29
- TPMODIFY 20, 24, 29
- transaction program
  - for the ANAME server 26
  - for the APING server 22
  - modifying the AFTP TP profile 20
  - modifying the ANAMED TP profile, MVS 29
  - modifying the APINGD TP profile 24
  - sample for the ANAME server 26
  - sample for the APING server 22
  - TPADD 22, 24, 26, 29
  - TPMODIFY 24, 29

## U

- updating the ACOPY EXEC on MVS 16
- updating the AFTP EXEC, MVS 15

## V

- VBUILD definition statement 29
- virtual machine, defining ANAME server's
  - creating a logon profile 10
  - creating CP directory 9
  - privilege class 8
  - virtual storage 8
- virtual machine, defining A3270 server's
  - creating a logon profile 12
  - creating CP directory 11
  - privilege class 11
  - virtual storage 11
- VM configuration tasks
  - A3270 server 10
    - adding a transaction program definition, AFTP 4
    - adding side information, AFTP 4
    - adding side information, ANAME 7
    - adding side information, APING 6
  - AFTP clients and servers 3
  - ANAME clients 7
  - ANAME server 8
  - APING clients and servers 5

VM configuration tasks *(continued)*  
   create a logon profile, A3270 server 12  
   create a logon profile, ANAME server 10  
   creating CP directory for A3270 server 11  
   creating CP directory for ANAME server 9  
   defining the A3270 server's virtual machine  
     creating a logon profile 12  
     creating CP directory 11  
   defining the ANAME server's virtual machine  
     creating a logon profile 10  
     creating CP directory 9  
   defining the APPC Application Suite 3  
   estimating VM storage requirements for ANAME  
     database 9  
   locating the A3270 server files 10  
   locating the AFTP files 3  
   locating the ANAME client REXX EXEC 7  
   locating the ANAME server files 8  
   locating the APING files 5  
   prerequisites 3  
   setting up the A3270 initialization file, VM 41  
   setting up the AFTP initialization file 35  
   setting up the ANAME initialization file 38  
   system software requirements 3  
   updating the logon profile, AFTPD 5  
   updating the logon profile, APINGD 7  
 VM samples  
   A3270 initialization file 42  
   AFTP initialization file 35, 37  
   ANAME initialization file 40  
   side information for AFTP 4  
   side information for ANAME 7  
   side information for APING 6  
   transaction program definition, AFTPD 4





---

# Readers' Comments — We'd Like to Hear from You

**z/OS Communications Server  
APPC Application Suite Administration  
Version 1 Release 2**

**Publication No. SC31-8835-00**

**Overall, how satisfied are you with the information in this book?**

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**How satisfied are you that the information in this book is:**

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Please tell us how we can improve this book:**

Thank you for your responses. May we contact you?  Yes  No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

---

Name

---

Address

---

Company or Organization

---

Phone No.



Fold and Tape

Please do not staple

Fold and Tape



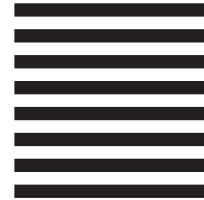
NO POSTAGE  
NECESSARY  
IF MAILED IN THE  
UNITED STATES

# BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation  
Software Reengineering  
Department G71A/ Bldg 503  
Research Triangle Park, NC  
27709-9990



Fold and Tape

Please do not staple

Fold and Tape





Program Number: 5694-A01



Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

SC31-8835-00



Spine information:



*z/OS Communications Server*

*z/OS V1R2.0 CS: APPC Application Suite  
Administration*

*Version 1  
Release 2*