APPENDIX B

CiscoSecure UNIX Server File Formats and Syntax

This appendix contains the following sections:

- Control File
- Message Catalogs
- AA Database
- Router Configurations

Control File

The following is an example CiscoSecure server control file:

```
/*
 * CiscoSecure UNIX Server Example Control File
 */

/* The license for this server. Multiple license keys may
 * be specified, each separated by a comma.
 */
LIST config_license_key = {"061db8afcf66db981f3c" }; /* */

/* The pathname of the Configuration Database */
LIST config_aa_database_filename = {"./configuration.database"};

/*
 * The pathname of the accounting log file - if this
 * variable is not specified then accounting information is not recorded.
 */
STRING config_accounting_database_filename = "/tmp/acct";

/* Default logging configuration - basic information is logged */
```

```
NUMBER config_logging_configuration = 0x7e;

/*
 * The maximum number of seconds to hold accounting information before
 * writing it to the accounting file (if specified).
 */
NUMBER config_accounting_write_frequency = 5;

/*
 * Per-NAS configuration records. The default encryption key is "arachnid"
 * for all NAS except 'boggle', which is using "heliotrope".
 */
NAS config_nas_config = {
  {
    "",     /* NAS name */
    "arachnid",/* secret key */
    "cat_1",/* message_catalogue_filename */
    1,      /* Authentication: username retries */
    3,      /* Authentication: password retries */
    1,      /* is default NAS configuration */
    1,      /* trusted NAS for SENDPASS */
    30      /* Password expiry period in days */
  },
  {
    "boggle",/* NAS name */
    "heliotrope",/* secret key */
    "",     /* message_catalogue_filename */
    2,      /* Authentication: username retries */
    2,      /* Authentication: password retries */
    0,      /* is not the default NAS configuration */
    0,      /* not a trusted NAS for SENDPASS */
    10      /* Password expiry period in days */
  }
};
```

Message Catalogs

Message catalogs allow system administrators to redefine the set of messages sent by the CiscoSecure server to the users connecting to a particular network access server. The following list identifies the default message IDs, message names, and message strings used by the CiscoSecure UNIX Server software.

```
0 AUTHEN_CLIENT_LOGIN_PROMPT    “\nUser Access Verification\n”
1 AUTHEN_CLIENT_USERNAME_PROMPT       “Username: “
2 AUTHEN_CLIENT_PASSWORD_PROMPT       “Password: “
3 AUTHEN_CLIENT_SIGN_ON_MESSAGE       “”
4 AUTHEN_CLIENT_CHANGEPASS_INTRO  “Change password sequence”
5 AUTHEN_CLIENT_PASSWORDS_IDENTICAL  “Error - passwords the same”
6 AUTHEN_CLIENT_PASSWORD_EXPIRED  “Your password has expired”
7 AUTHEN_CLIENT_TOO_MANY_TRIES_FOR_USERNAME  “Too many tries for
username”
8 AUTHEN_CLIENT_TOO_MANY_TRIES_FOR_PASSWORD  “Too many tries for
password”
9 AUTHEN_CLIENT_NEW_PASSWORD1   “New password: “
10 AUTHEN_CLIENT_NEW_PASSWORD2   “New password again: “
11 AUTHEN_CLIENT_PASSWORDS_DIFFERENT  “The passwords are different”
12 AUTHEN_CLIENT_BAD_PASSWORD   “Bad password”
13 AUTHEN_CLIENT_CANT_CHANGE_PASSWORD  “You cannot change your
password”
14 AUTHEN_CLIENT_ACCOUNT_EXPIRY_WARNING  “Your account will expire in
%d days”
15 AUTHEN_CLIENT_PASSWORD_EXPIRY_WARNING   “Your password will expire
in %d days”
16 AUTHEN_CLIENT_NEW_PASSWORD_CRITERIA “A password must be between six
and thirteen characters, containing at least one alphabetic and
numeric character.”

18 AUTHEN_USER_NOT_FOUND        “Authentication - User not found”
19 AUTHEN_BAD_METHOD_FOR_USER  “Authentication - Bad method for user”
20 AUTHEN_BAD_TYPE              “Authentication - Bad type”
21 AUTHEN_NO_USERNAME        “Authentication - No username specified”
22 AUTHEN_INSUFFICIENT_PRIVILEGE  “Authentication - Insufficient
privilege”
23 AUTHEN_UNEXPECTED_DATA       “Authentication - Unexpected data”
24 AUTHEN_UNEXPECTED_RESERVED_DATA  “Authentication - Unexpected
reserved data”
25 AUTHEN_INCORRECT_PASSWORD    “Authentication - Incorrect password”
26 AUTHEN_ABORTED_SEQUENCE      “Authentication - Aborted sequence”
27 AUTHEN_FILEHANDLING_ERROR   “Authentication - File handling error”
```

```
28 AUTHEN_UNKNOWN_PASSWORD_TYPE  “Authen - Unknown password type”
29 AUTHEN_USER_NOT_IN_FILE      “Authentication - User not in file”
30 AUTHEN_ERROR_IN_EXTERNAL_FN,”Authentication - Error in external
function”
31 AUTHEN_BAD_SERVICE           “Authentication - Bad Service”
32 AUTHEN_BAD_ACTION            “Authentication - Bad Action”
33 AUTHEN_SENDPASS_OK           “Authentication - SENDPASS (ok)”
34 AUTHEN_SENDPASS_FAIL         “Authentication - SENDPASS (fail)”

35 PROTOCOL_USERNAME_TOO_LONG   “Protocol - Username too long”
36 PROTOCOL_NASNAME_TOO_LONG    “Protocol - NAS name too long”
37 PROTOCOL_NASPORT_TOO_LONG    “Protocol - NAS port name too long”
38 PROTOCOL_NACADDR_TOO_LONG    “Protocol - NAC address too long”
39 PROTOCOL_BAD_PRIVILEGE       “Protocol - Invalid privilege field”
40 PROTOCOL_ACTIVE_SESSION      “Protocol - Session id in use”
41 PROTOCOL_NO_SESSION          “Protocol - No session found”
42 PROTOCOL_INCORRECT_TYPE      “Protocol - Incorrect type”
43 PROTOCOL_INCORRECT_SESSION   “Protocol - Incorrect session”
44 PROTOCOL_INCORRECT_SEQUENCE  “Protocol - Incorrect sequence”
45 PROTOCOL_INCORRECT_VERSION   “Protocol - Incorrect version”
46 PROTOCOL_GARBLED             “Protocol - Garbled message”
47 PROTOCOL_READ_TIMEOUT        “Protocol - Read timeout”
48 PROTOCOL_CONNECTION_CLOSED   “Protocol - Connection closed”
49 PROTOCOL_BAD_TYPE            “Protocol - Bad type”
50 PROTOCOL_MAX_USERS_EXCEEDED  “Maximum number of users exceeded”
51 PROTOCOL_ENCRYPTION_MISMATCH  “Mismatched encryption”

52 AUTHOR_NO_SERVICE          “Authorization - No service specified”
53 AUTHOR_FAILED_MANDATORY_ARG  “Authorization - Failed mandatory
argument”
54 AUTHOR_FAILED_COMMAND_LINE   “Authorization - Failed command line”
55 AUTHOR_FAILED_SERVICE        “Authorization - Failed service”
56 AUTHOR_FAILED_TIME           “Authorization - Failed time
qualification”
57 AUTHOR_BAD_ARGUMENT          “Authorization - Bad argument”
58 AUTHOR_NO_COMMAND          “Authorization - No command specified”
59 AUTHOR_FAILED_CMD            “Authorization - Failed command”
60 AUTHOR_NO_PROTOCOL           “Authorization - No protocol”
61 AUTHOR_UNKNOWN_USER          “Authorization - Unknown user”
62 AUTHOR_INVALID_NAS_OR_PORT   “Authorization - Unauthorized NAS or
PORT”
63 AUTHOR_COMMAND_AUTHORIZED    “Authorization - Command authorized”
```

AA Database

This section provides an example CiscoSecure AA database file and the detailed grammar that the database follows.

AA Database File

The following is an example CiscoSecure AA database file:

```
# Sample AA Database
#
# Unless otherwise specified, the following are applied:
#       default service = deny
#       default protocol = deny
#       default attribute = deny
#       default cmd = deny
#

default = {
  # The default password is the one used by the system (i.e. /etc/passwd)
  password = system
  set autocmd = “telnet registration-server”
}

group = staff {
  password = des “sefjKaLm7zybE”
  privilege = clear “operator” 2
}

# The admin group is also a member of ‘staff’ and can enable to
# privilege level 15 with a DES password (a UNIX encrypted password.)
group = admin {
  member = staff
  password = des “Aj2pwjbnZlsoh”
  privilege = des “sefjKaLm7zybE” 15
}

# Fred uses an ‘skey’ password, is a member of ‘admin’, and can do
anything.
user = fred {
  member = admin
  password = skey
  password = chap “fred-chap”  # Fred’s CHAP password
```

```
  default service = permit
  default protocol = permit
  default cmd = permit
  default attribute = permit
}

# Joy can use any PPP protocol except IPX and can issue any exec command
# except the ‘enable’ command.
user = joy {
  member = staff
  password = clear “My ClearText Password”
  service = ppp {
    default protocol = permit
    prohibit protocol = ipx
  }
  service = shell {
    default cmd = permit
    prohibit cmd = enable
  }
}

# Tom can only run PPP/IP on NAS0 - NAS9, any tty port.
# However, he can only run IPX on NAS12, any tty port.
user = tom {
  service = ppp {
    protocol = ip {
      allow “nas[0-9]” “tty.*”
    }
    protocol = ipx {
      refuse “nas12” “tty.*”
    }
}

# Ralph can run an EXEC, but he falls into the ‘default’ above,
# which specifies an autocommand.  His account is valid after 1 January
1996,
# and expires on 31 December 1996.
user = ralph {
  service = shell {
  }
  valid = “1 Jan 96”
  expires = “31 Dec 96”
}

# Frank can only start an EXEC session at night and on weekends.
```

```
user = frank {
  service = shell {
    default cmd = permit
    default attribute = permit
    time = Any 2300 - 0559
    time = Sat, Sun 0000 - 2359
  }
}

# Joe can run an EXEC, but only for the one year period specified.
user = joe {
  service = shell {
    # No cmd defaults to EXEC
  } from “1 Jan 96” until “31 Dec 96”
}

# Robert can start PPP/IP, but only if his machine uses IP address
# 131.108.12.3, and with the input and output access lists specified.
user = robert {
  service = ppp {
    protocol = ip {
      set addr = 131.108.13.3
      set inacl = 103
      set outacl = 105
    }
  }
}

# Anita can run PPP/IP, but gets an address from an IP address
# pool named ‘alternet’ on the NAS.  She may only connect on NAS1,
# any tty port.
user = anita {
  service = ppp {
    protocol = ip {
      set addr-pool = alternet
      allow “nas1” “tty.*”
    }
  }
}

# Sam can use ARAP, which may have an Appletalk access list 601
# applied.
user = sam {
  service = arap {
    set optional acl = 601
```

```
  }
}

# Bob can only issue the exec command “show users”
user = bob {
  service = shell {
    cmd = show {
      permit “users”
    }
  }
}

# Rob can only telnet only to hosts with IP addresses from
# 131.101.13.2 - 131.101.13.254 and issue any ‘show’ commands.
user = rob {
  service = shell {
    cmd = telnet {
      deny “131\.101\.13\.1”
      permit “131\.101\.13\.[0-9]+”
    }
    cmd = show {
      permit “.*”
    }
  }
}
```

AA Database Grammar

The AA database grammar accurately defines the syntax of the AA database.

Grammar

The AA database conforms to the following grammar:

```
config                    :=  <empty>
                          |    config declaration

declaration               :=  USER ‘=’ string ‘{‘ attribute_list time_qualifier ‘}’
                          |   GROUP ‘=’ string ‘{‘ attribute_list time_qualifier ‘}’
                          |   DEFAULT ‘=’ ‘{‘ attribute_list ‘}’

attribute_list            :=  <empty>
```

```
                          |   attribute_list attribute

attribute                 :=  unqualified_attribute
                          |   qualified_attribute date_qualifier

unqualified_attribute     :=  MEMBER ‘=’ string
                          |   DEFAULT SERVICE ‘=’ permission
                          |   PROHIBIT SERVICE ‘=’ string
                          |   ACCOUNTING ‘=’ string
                          |   PROHIBIT ACCOUNTING
                          |   qualification
                          |   service
                          |   setting

qualified_attribute       :=  PASSWORD ‘=’ password
                          |   PRIVILEGE ‘=’ password NUMERIC
                          |   SET attribute_value_pair

service                   :=  SERVICE ‘=’ PPP ‘{‘ ppp_protocol_authorization
                              time_qualifier’}’
                          |   SERVICE ‘=’ SHELL ‘{‘ shell_command_authorization
                              time_qualifier ‘}’
                          |   SERVICE ‘=’ string ‘{‘ general_block_authorization
                              time_qualifier ‘}’

ppp_protocol_authorization  :=  <empty>
                          |   ppp_protocol_authorization protocol_authorization
                          |   ppp_protocol_authorization service_attribute_value_pair
                          |   ppp_protocol_authorization setting
                          |   ppp_protocol_authorization DEFAULT PROTOCOL ‘=’ permission
                          |   ppp_protocol_authorization PROHIBIT PROTOCOL ‘=’ string

shell_command_authorization     := <empty>
                          |   shell_command_authorization command_authorization
                          |   shell_command_authorization service_attribute_value_pair
                          |   shell_command_authorization setting
                          |   shell_command_authorization DEFAULT CMD ‘=’ permission
                          |   shell_command_authorization PROHIBIT CMD ‘=’ string

general_block_authorization :=  <empty>
                          |   general_block_authorization service_attribute_value_pair
                          |   general_block_authorization setting

protocol_authorization    :=  PROTOCOL ‘=’ string  ‘{‘ service_attribute_list
                              time_qualifier ‘}’
```

```
command_authorization       :=  CMD ‘=’ string ‘{‘ command_match time_qualifier ‘}’

password                    :=  FILESPEC string
                            |   SYSTEM
                            |   NO_PASSWORD
                            |   DES string
                            |   CLEAR string
                            |   ARAP string
                            |   CHAP string
                            |   PAP string
                            |   STRING
                            |   STRING string

command_match               :=  <empty>
                            |   command_match permission string
                            |   command_match setting
                            |   command_match service_attribute_value_pair
                            |   command_match DEFAULT CMD ‘=’ permission

service_attribute_list      :=  <empty>
                            |   service_attribute_list service_attribute_value_pair
                            |   service_attribute_list setting

service_attribute_value_pair:=  SET attribute_value_pair date_qualifier

attribute_value_pair        :=  STRING ‘=’ value
                            |   OPTIONAL STRING ‘=’ value

setting                     :=  DEFAULT ATTRIBUTE ‘=’ permission
                            |   PRE_PROCESS ‘=’ string
                            |   POST_PROCESS ‘=’ string
                            |   PROHIBIT PRE_PROCESS
                            |   PROHIBIT POST_PROCESS
                            |   filter string string

time_qualifier              :=  <empty>
                            |   time_qualifier TIME ‘=’ day_list HOUR ‘-’ HOUR

day_list                    :=  WEEKDAY
                            |   day_list ‘,’ WEEKDAY

qualification               :=  EXPIRES ‘=’ QSTRING
                            |   VALID ‘=’ QSTRING
```

```
value                    :=  string
                         |   NUMERIC

string                   :=  STRING
                         |   QSTRING

permission               :=  PERMIT
                         |   DENY

filter                   :=  ALLOW
                         |   REFUSE

date_qualifier           :=  <empty>
                         |   date_qualifier FROM QSTRING
                         |   date_qualifier UNTIL QSTRING
```

Lexical Conventions

The AA database conforms to the following lexical conventions:

- Newline characters are implementation-specific. By convention, UNIX uses 0x0A; DOS and related systems use 0x0A 0x0D.
- Lines including the # symbol are treated as comments. All characters on the line from the # up to and including the newline character (\n), are ignored.
- Space and tab are treated as white-space separator characters, except where they are enclosed in double quotation marks.
- Statements must be separated by the newline character.
- Continuation lines are indicated by the backslash (\) immediately preceding a newline character. Strings may contain the continuation character, unless it immediately precedes the new line. Under these circumstances, the string must be enclosed in double quotation marks.
- String is any combination of alphanumeric and other characters, excluding punctuation characters, and is not matched by any defined keystring.
- Qstring is a string enclosed in quotation marks. All printable characters except included quotation marks are accepted.

Router Configurations

This section contains two router configuration files. The router “left” is using CiscoSecure UNIX Server software for authentication and authorization. The router “right” is using internal passwords for authentication and authorization. (The terms “left” and “right” are merely the names of two example routers, and completely unrelated to any directional instruction for a router configuration.) In this example, a synchronous serial interface is used for connection between the two routers.

```
! left.cfg - A router using CiscoSecure for Authentication
! and Authorization
version 11.0
service udp-small-servers
service tcp-small-servers
!
hostname left
!
aaa new-model
aaa authentication login admin tacacs+ enable
aaa authentication login vty line
aaa authentication ppp customer tacacs+
aaa authorization commands 1 tacacs+
aaa authorization commands 15 tacacs+ if-authenticated
enable password san-fran
!
username right password 7 15101E0E062B392D2F3B21
username left password 7 070D344E4C0815001106
!
!
interface Loopback3
 ip address 204.176.108.1 255.255.255.0
!
interface Ethernet0
 ip address 151.50.128.1 255.255.254.0
 media-type 10BaseT
 bridge-group 1
!
interface Ethernet1
 ip address 151.50.144.1 255.255.255.0
 media-type 10BaseT
 bridge-group 1
!
interface Serial0
 ip address 151.50.16.1 255.255.255.252
```

```
 encapsulation ppp
 bandwidth 125000
 fair-queue 64
 ppp authentication chap customer
!
interface Serial1
 no ip address
 shutdown
!
interface TokenRing0
 ip address 151.50.160.1 255.255.255.128
 ring-speed 4
 bridge-group 1
!
interface TokenRing1
 no ip address
 shutdown
!
tacacs-server host 151.50.129.9
tacacs-server key arachnid
!
banner exec ^C

Left Router - Using TACACS+

^C
!
line con 0
 password cisco
 login authentication admin
line aux 0
 transport input all
line vty 0 4
 password cisco
 login authentication admin
!
end



! right Router - Using internal Authentication and Authorization
version 11.0
service udp-small-servers
service tcp-small-servers
!
```

```
hostname right
!
enable password san-fran
!
username right password 7 13070210090D
username left password 7 020411590907
username right
!
!
interface Ethernet0
 ip address 144.251.100.201 255.255.255.0
 shutdown
!
interface Ethernet1
 no ip address
 shutdown
 media-type 10BaseT
!
interface Serial0
 ip address 151.50.16.2 255.255.255.252
 encapsulation ppp
 bandwidth 125000
 clockrate 125000
 ppp authentication chap
!
interface Serial1
 no ip address
 shutdown
 clockrate 125000
!
interface TokenRing0
 no ip address
 shutdown
!
interface TokenRing1
 no ip address
 shutdown
!
banner exec ^C

right Router - Using internal information for AA

^C
!
line con 0
```

```
 password cisco
 login
line aux 0
 transport input all
line vty 0 4
 password cisco
 login
!
end
```