Lucent Technologies
Bell Labs Innovations

MAX TNT®

True Access™ Operating System (TAOS) Addendum

Part number: 7820-0502-008
For software version: 9.0
December 2000

Notice

Every effort was made to ensure that the information in this document was complete and accurate at the time of printing, but information is subject to change.

Safety, Compliance, and Warranty Information

Before handling any Lucent Access Networks hardware product, read the *Edge Access Safety and Compliance Guide* included in your product package. See that guide also to determine how products comply with the electromagnetic interference (EMI) and network compatibility requirements of your country. See the warranty card included in your product package for the limited warranty that Lucent Technologies provides for its products.

Security Statement

In rare instances, unauthorized individuals make connections to the telecommunications network through the use of access features.

Trademarks

Copyrights for Third-Party Software Included in Lucent Access Networks Software Products

Ordering Information

You can order the most up-to-date product information and computer-based training online at http://www.lucent.com/ins/bookstore.

Feedback

Lucent Technologies appreciates your comments, either positive or negative, about this manual. Please send them to techpubs@ascend.com.

Lucent Technologies

Customer Service

To obtain product and service information, software upgrades, and technical assistance, visit the eSight™ Service Center at `http://www.esight.com`. The center is open 24 hours a day, seven days a week.

Finding information and software

The eSight Service Center at `http://www.esight.com` provides technical information, product information, and descriptions of available services. Log in and select a service. The eSight Service Center also provides software upgrades, release notes, and addenda. Or you can visit the FTP site at `ftp://ftp.ascend.com` for this information.

Obtaining technical assistance

The eSight™ Service Center at `http://www.esight.com` provides access to technical support. You can obtain technical assistance through email or the Internet, or by telephone.

If you need to contact Lucent Technologies for assistance, make sure that you have the following information available:

- Active contract number, product name, model, and serial number
- Software version
- Software and hardware options
- If supplied by your carrier, service profile identifiers (SPIDs) associated with your line
- Your local telephone company's switch type and operating mode, such as AT&T 5ESS Custom or Northern Telecom National ISDN-1
- Whether you are routing or bridging with your Lucent product
- Type of computer you are using
- Description of the problem

Obtaining assistance through email or the Internet

If your services agreement allows, you can communicate directly with a technical engineer through Email Technical Support or eSight Live chat. Select one of these sites when you log in to `http://www.esight.com`.

Calling the technical assistance center (TAC)

If you cannot find an answer through the tools and information on eSight or if you have a very urgent need, contact TAC. Access the eSight Service Center at `http://www.esight.com` and click `Contact Us` below the Lucent Technologies logo for a list of telephone numbers inside and outside the United States.

You can alternatively call (800) 272-3634 for a menu of Lucent services, or call (510) 769-6001 for an operator. If you do not have an active services agreement or contract, you will be charged for time and materials.

Table of Contents

About this addendum

The True Access™ Operating System (TAOS) contains a foundation of built-in software features for WAN access environments, as well as optional extensions that require separate licensing to support a wide variety of WAN access environments.

This addendum describes all new features and extensions that have been introduced for MAX TNT™ units since TAOS 8.0.2. Some of the features have been introduced in earlier TAOS 8.x releases.

The section “MAX TNT upgrade and downgrade procedures” in the *MAX TNT 9.0 Release Note* describes how to upgrade and downgrade your system software.

Caution: You must use the software procedures in the “MAX TNT upgrade and downgrade procedures” in the *MAX TNT 9.0 Release Note* to load this TAOS 9.0 onto your system or restore a previous version. Read the instructions carefully before upgrading or dowgrading your system.

How to use this addendum

The Table of Contents on page v and the tables in the following section list the TAOS features in this release. If you are reading this addendum in PDF format, you can click the feature name to go directly to the feature.

For information about obtaining the software described in this addendum, see the section “Obtaining the MAX TNT TAOS 9.0 software” in the *MAX TNT TAOS 9.0 Release Note*.

TAOS built-in features

Table 1 through Table 4 show the built-in features that have been added to TAOS since the last major release.

Table 1. MAX TNT TAOS 9.0 WAN access server features

Feature	Introduced in
Support for the Ethernet-3-ND slot card	TAOS 9.0
Support for the E3-ATM slot card	TAOS 8.x
Support for the PCTFI slot card	TAOS 9.0
Support for port-speed configuration on Ethernet-3 slot cards	TAOS 9.0
ISDN PRI support for the STM-0 slot card	TAOS 9.0
Traffic shaping on DS3-ATM2 and E3-ATM slot cards	TAOS 9.0

Table 2. MAX TNT TAOS 9.0 modem manager features

Feature	Introduced in
Firmware versions for digital modems	TAOS 9.0
Firmware versions for MultiDSP cards	TAOS 9.0
Support for HDLC-NRM on Series56 III Digital Modem cards	TAOS 9.0
Support for V.110 subrates for MultiDSP cards	TAOS 9.0
Support for progress code 33	TAOS 9.0

Table 3. MAX TNT TAOS 9.0 authentication and accounting client features

Feature	Introduced in
Support for CHAP name challenge during incoming calls	TAOS 9.0
Nonauthentication option for asynchronous framed users	TAOS 9.0
Service-Type (6) set to Call-Check for CLID and DNIS authentication	TAOS 8.x
Service-Type attribute (6) in RADIUS accounting records	TAOS 9.0
RADIUS: New authentication-delay attribute	TAOS 9.0
Stripping portions of the username from RADIUS access requests	TAOS 9.0
CLID and DNIS authentication cause codes	TAOS 8.X
16-bit vendor-specific attribute (VSA) support	TAOS 9.0

Table 4. MAX TNT TAOS 9.0 management agent features

Feature	Introduced in
SNMP: Increased number of SNMP managers	TAOS 9.0
SNMP: MIB support for DS3-ATM version 2 and E3-ATM slot cards	TAOS 9.0
SNMP: DS3 MIB support for the channelized T3 slot card	TAOS 9.0
SNMP: Support for SMIv2 syntax	TAOS 9.0
SNMP: Support for the L2TP MIB	TAOS 9.0
SNMP: Support for the Remote Ping MIB	TAOS 9.0
SNMP: Support for the ISDN Type of Number MIB	TAOS 9.0
SNMP: Support for SNMPv3 USM privacy	TAOS 9.0
SNMP: SNMP manager support for the USM MIB	TAOS 9.0
SNMP: Support for SNMPv3 notifications	TAOS 9.0
SNMP: Saving and restoring encrypted configurations	TAOS 9.0
SNMP: Support for the ifStackTable in the IF-MIB	TAOS 9.0

Table 4. MAX TNT TAOS 9.0 management agent features (continued)

Feature	Introduced in
SNMP: WAN line table now shows signaling type for T1 and E1 lines	TAOS 9.0
Support for multiple requests in a call-logging packet	TAOS 9.0
SNMP: Support for the NoResourceAvailable trap	TAOS 9.0
SNMP: Enhanced support for the sysSlotStateChange trap	TAOS 9.0
SNMP: VoIP call jitter reporting	TAOS 9.0
SNMP: Support for VoIP call logging	TAOS 9.0
Support for encryption of configuration transferred through TFTP	TAOS 9.0
Support for a maintenance state for slot cards	TAOS 9.0
NavisAccess: Enhanced network management	TAOS 9.0
Rlogin and raw TCP support added to terminal-server menus	TAOS 9.0
New Diag command	TAOS 9.0
New options for the NSLookup command	TAOS 9.0
Displaying call session and authentication statistics	TAOS 9.0
Support for the Load Tar command using multiple filenames	TAOS 9.0

TAOS extensions

The following extension features been added to TAOS since the last major release. Extension features are available when the appropriate software license has been enabled.

Table 5. MAX TNT TAOS 9.0 global digital access extension features

Feature	Introduced in
WORM-ARQ for personal digital cellular phones	TAOS 8.x
Support for PIAFS 2.1 on the 96-port MultiDSP card	TAOS 9.0
Rejecting collect calls on Brazilian R2 signaling lines	TAOS 9.0

Table 6. MAX TNT TAOS 9.0 Signaling System 7 (SS7) extension features

Feature	Introduced in
Q.931+ for PacketStar SS7 signaling gateways	TAOS 8.x
SS7 Q.931 messaging support for V.110 calls	TAOS 9.0
SS7 IPDC RTE for congestion control (Lucent Technologies Softswitch only)	TAOS 8.x

Table 6. MAX TNT TAOS 9.0 Signaling System 7 (SS7) extension features (continued)

Feature	Introduced in
SS7 IPDC: Reporting VoIP call statistics	TAOS 9.0
SS7: PRI tunneling in IPDC (IPDC 0.15)	TAOS 9.0
SS7 gateway IPDC support for E1 trunks	TAOS 9.0
SS7 continuity checks for E1 lines	TAOS 9.0

Table 7. MAX TNT TAOS 9.0 MultiVoice™ extension features

Feature	Introduced in
NavisAccess support for VoIP call reporting	TAOS 9.0
Storing voice announcements in the FAT-16 flash memory file system	TAOS 9.0
MultiVoice: Compress a two-frame VoIP packet into three ATM cells	TAOS 9.0
MultiVoice: Support of Full Rate GSM audio codec	TAOS 9.0
MultiVoice: Support for G.729-encoded voice announcement files	TAOS 9.0
MultiVoice: Arbitrary announcement playback and tone collection	TAOS 9.0
Deactivating trunks used for VoIP calls	TAOS 9.0
MultiVoice: Support for CLID substitution and early-ringback	TAOS 9.0
MultiVoice: Support for T.38 and transparent fax/modem for IPDC	TAOS 9.0
MultiVoice: Support for transparent fax/modem over VoIP	TAOS 9.0
MultiVoice: Support for modem and VoIP cohabitation	TAOS 9.0
Jitter buffer and packet redundancy for real-time fax operations	TAOS 9.0
MultiVoice: Support for real-time fax backward compatibility	TAOS 9.0
MultiVoice: Real-time fax maximum data transmission rate limit	TAOS 9.0
MultiVoice: Trunk prefixing	TAOS 9.0
MultiVoice: DTMF tone processing over R2 signaling	TAOS 9.0
Support for A-Law companding on DTMF DSP code	TAOS 9.0
MultiVoice: Support for E1 R2 variable-length DNIS without EOP	TAOS 9.0
MultiVoice: Support for Feature Group D (FGD)	TAOS 9.0
MultiVoice: Support for multiple logical gateways	TAOS 9.0
MultiVoice: Report trunk capacity to the gatekeeper	TAOS 8.x
E1 CMF R2: Collect 15-digit dial strings	TAOS 9.0
MultiVoice: Delay of charges until call is answered (true connect)	TAOS 9.0
MultiVoice: Support for configurable operator assistance	TAOS 9.0
MultiVoice: Support for sequential call dialing without authentication	TAOS 9.0

Table 7. MAX TNT TAOS 9.0 MultiVoice™ extension features (continued)

Feature	Introduced in
MultiVoice: Support for VoIP PSTN attributes	TAOS 9.0
MultiVoice: Support for IPDC RMCP and AMCP messages	TAOS 9.0

Table 8. MAX TNT TAOS 9.0 tunneling extension features

Feature	Introduced in
Proxy LCP and authentication for L2TP tunnels	TAOS 9.0
Support for Tunnel-Client-Auth-ID and Tunnel-Server-Auth-ID	TAOS 9.0
Support for tunnel assignment IDs	TAOS 9.0
Support for Tunnel-Private-Group-ID in L2TP	TAOS 9.0
Enhanced RADIUS accounting and call logging for VPNs	TAOS 9.0
Support for L2TP, PPTP, and L2F disconnect and progress codes	TAOS 9.0

Table 9. MAX TNT TAOS 9.0 virtual routing extension features

Feature	Introduced in
Support for VRouters with ATMP	TAOS 9.0
Support for VRouters in IPX networks	TAOS 9.0

Table 10. MAX TNT TAOS 9.0 short-duration transaction networks extension features

Feature	Introduced in
SDTN support for TPDU terminals	TAOS 9.0

Table 11. MAX TNT TAOS 9.0 IP fax extension feature

Feature	Introduced in
Support for the Atlas redialer and DID	TAOS 9.0

Features requests in TAOS 9.0

Table 12 lists the identification numbers for features requested for the MAX TNT.

Table 12. Feature requests in this release

Feature ID	Description
260605	Support for Tunnel-Client-Auth-ID and Tunnel-Server-Auth-ID
509955	MultiVoice: DTMF tone processing over R2 signaling
510056	WORM-ARQ for personal digital cellular phones
510154	New ATMP disconnect codes
510174	Rejecting collect calls on Brazilian R2 signaling lines
510183	Nonauthentication option for asynchronous framed users
510187	SS7 continuity checks for E1 lines
510223	Support for VRouters with ATMP
510228	Support for Tunnel-Client-Auth-ID and Tunnel-Server-Auth-ID
510247	SDTN support for TPDU terminals
510276	SNMP: Support for the ISDN Type of Number MIB
510281	Proxy LCP and authentication for L2TP tunnels
510288	Contents of the user-user IE now available to RADIUS
510300	Support for CHAP name challenge during incoming calls
510321	Service-Type (6) set to Call-Check for CLID and DNIS authentication
510322	Service-Type attribute (6) in RADIUS accounting records
510344	SNMP: Support for the L2TP MIB
510346	Support for tunnel assignment IDs

Notices and caveats

Notice of memory requirement in TAOS 9.0

To upgrade to MAX TNT TAOS 9.0.0, your MAX TNT unit must be equipped with the 32MB flash card. Please contact your Lucent sales representative to purchase the 32MB flash card.

Notice of parameter name changes in the External-Auth profile

In TAOS 8.0, the `dnis-password` and `clid-password` parameters were added to the `External_Auth` profile. With these parameters, you were able to set RADIUS passwords for DNIS and CLID preauthentication.

In TAOS 9.0, the `dnis-password` and `clid-password` parameters were moved to the `password` subprofile of the `External-Auth` profile. The parameter names were also changed, as shown in the following sample subprofile (shown with default values):

```
[in EXTERNAL-AUTH:password-profile]
clid = Ascend-CLID
dnis = Ascend-DNIS
```

If your unit is configured with DNIS and CLID passwords, after upgrading from TAOS 8.x to TAOS 9.0, the unit will no longer recognize the `dnis-password` and `clid-password` values that were set in prior releases and dial-in users might experience a busy tone.

To restore the DNIS and CLID preauthorization passwords, you must apply the value of the `dnis-password` and `clid-password` parameters (set in earlier TAOS 8.x releases), to the new `dnis` and `clid` parameters as follows:

```
admin> read external-auth
EXTERNAL-AUTH read

admin> set password-profile dnis = secretdnis

admin> set password-profile clid = secretclid

admin> write
EXTERNAL-AUTH written
```

Notice of change in supported values for the signaling mode

After you upgrade a MAX TNT that is configured as a MultiVoice gateway, the MAX TNT unit might generate a “bad value” error message for the value assigned to the Signaling-Mode parameter in the Line-Config subprofile of T1 profile. This situation occurs when you upgrade the MAX TNT from either of the following limited availability releases to TAOS 9.0:

- TAOS 8.0-103.x
- TAOS 8.0-118.x

When these two limited availability releases were compiled, the supported values for the Signaling-Mode parameter were defined as enumerated values, rather than hardcoded values, as is done for TAOS 9.0. Applying a saved configuration from either limited availability release to TAOS 9.0 causes the “bad value” error.

To correct this error, you must reset the value of the Signaling-Mode parameter after applying the saved configuration and reinitializing the MAX TNT unit.

Notice about MultiDSP cards

In TAOS 9.0, you can now combine 48-port and 96-port MultiDSP cards in the a MAX TNT unit for V.90 and ISDN dial-up termination.

Notice of support for a new shelf controller

With TAOS 8.0.3, Lucent introduced support for a new shelf-controller hardware implementation (model number TNT-SP-SC-SS). The backplane of the new shelf controller does not include multishelf components, so it does not support the physical connection of multiple chassis to operate as one virtual unit. All other functionality is identical with the older shelf controller (model number TNT-SP-SC).

To use the new shelf controller, the unit must be running TAOS 8.0.3 or later.

Caution: The new shelf controller does not power up if it is installed in a unit running an earlier version of TAOS. When the new shelf controller has been installed, you cannot downgrade the unit to software earlier than TAOS 8.0.3.

Notice of modified behavior during IPCP negotiation

With TAOS 9.0, the MAX TNT unit requires a valid System-IP-Addr setting to complete IPCP negotiation. For example, the following commands explicitly set the system address to the shelf controller IP address:

```
admin> get ip-int { {1 c 1} 0} ip-address
ip-address = 10.2.3.4

admin> read ip-global
IP-GLOBAL read

admin> set system-ip-addr = 10.2.3.4

admin> write
IP-GLOBAL written
```

Note: If the System-IP-Addr setting is null, the system terminates PPP connections during the IPCP negotiation phase.

Notice about default settings in the call-logging profile

In the Call-Logging profile, the default settings for the Call-Log-Limit-Retry and Call-Log-Timeout parameters are as follows:

```
[in CALL-LOGGING]
call-log-limit-retry = 0
call-log-timeout = 1
```

With call logging enabled and with these parameters left to their default values, if the connection to the call-logging host (such as a RADIUS accounting server or NavisAccess™) fails, the MAX TNT unit continues to send Start and Stop packets to the call-logging host indefinitely.

The setting of `0` (zero) for the Call-Log-Limit Retry parameter indicates an unlimited number of retries. A setting of `1` (one) for Call-Log-Timeout indicates that the MAX TNT unit waits only 1 second before retrying.

To avoid unlimited retrys, Lucent recommends that you change the default settings of the Call-Log-Limit-Retry parameter from `0` to `2` or `3` and the Call-Log-Timeout parameter from 1 to between `5` and `10` seconds. For example:

```
admin> read call-logging
CALL-LOGGING read

admin> set call-log-limit-retry = 3

admin> set call-log-timeout = 10

admin> write
CALL-LOGGING written
```

Notice about a change in the terminal server password length

TAOS 7.2.3 reduced the maximum allowable password length for the Password-For-Direct-Access parameter from 64 to 21 characters. The current version of the *APX 8000/DSLTNT/MAX TNT Reference* indicates a maximum value of 64 characters.

Notice of TAOS interoperability with SS7 signaling software

The MAX TNT supports two separate software licenses for integrating the units into Signaling System 7 (SS7) networks:

- Access SS7 Gateway Control Protocol (ASGCP). This method of integration enables the MAX TNT to terminate data calls in an SS7 network. The signaling gateway must be ICD for softswitch (formerly ASG). ICD stands for Internet Call Diversion.
- IP Device Control (IPDC). IPDC is a third-party proprietary protocol. This method of integration enables the MAX TNT to terminate both voice and data calls. The signaling gateway can be ICD for softswitch or Lucent Softswitch.

TAOS 9.0 supports interoperability with sections of IP Device Control (IPDC) 0.15.1, including PRI tunneling.

Notice of discontinuance of configurable RADIUS port and ID space

In TAOS 8.x, the default settings for User Datagram Protocol (UDP) source ports and ID spaces for communication with a RADIUS server specified the use of a unique source port for each card and a distinct ID space for both authentication and accounting requests. However, the unit could be configured to use a single source port and ID space system-wide, in order to accommodate certain RADIUS server daemons that had a system-unique requirement.

Because no known RADUIS servers continue to maintain this requirement, and because increased port density makes the use of a single port and ID space undesirable, with TAOS 9.0, the MAX TNT unit always uses port-unique source ports and always sends RADIUS authentication and accounting requests with distinct RADIUS IDs. The following parameters are therefore no longer supported and have been removed from the External-Auth profile:

```
[EXTERNAL-AUTH]
rad-id-space = distinct
rad-id-source-unique = port-unique
```

Note: The `rad-ip-space` and `rad-id-source-unique` parameters no longer appear in the External-Auth profile with TAOS 9.0. If you downgrade the unit to an earlier release, the parameters revert to their default values for that release.

Notice of discontinuance of software support

Software support has been discontinued for the MAX TNT Ethernet-0 slot card (TNT-SL-E10), the Fast (100 MB) Ethernet-1 slot card (TNT-SL-E100), and the older MAX TNT Hybrid Access slot cards (TNT-SL-HA128 and TNT-SL-HA192).

Notice of discontinuance of MAX TNT support for DSL

Support for digital subscriber loop (DSL) functionality is discontinued in MAX TNT units as of TAOS 8.0. The DSLTNT™ platform continues support for existing DSL products and will introduce additional DSL functionality in future releases.

Notice of deprecated management features

Use of the `if-admin` diagnostic command is deprecated. The functionality that was provided by the `-d` (down) and `-u` (up) options of the command is now provided by `read`, `set`, and `write` operations on one of the following profiles:

- The Admin-State-Perm-If profile for permanent interfaces such as a nailed interface
- The Admin-State-Phys-If profile for physical interfaces such as a T1 line

The other options of the `if-admin` command are not supported.

Use of the `call-log-radius-compat` parameter in the Call-Logging profile is deprecated in this software version.

The `callActiveIfIndex` and `callStatusIfIndex` objects in the call MIB are not supported in this software version.

This following objects are no longer supported in this software version:

- The `lmodem.mib`
- The `resetStat` group in `ascend.mib`
- The `consoleTable`, `doTable`, and `hostStatusTable` in `ascend.mib`

Notice about upgrading slot cards

If you replace a MAX TNT Fast (100 MB) Ethernet-1 slot card (TNT-SL-E100) with a newer Ethernet card (TNT-SL-E10-100 or TNT-SL-E100-V-C), you must write new Ethernet profiles for the new card. The old Ethernet profiles do not carry forward.

If you replace an older MAX TNT Hybrid Access slot card (TNT-SL-HA128 or TNT-SL-HA192) with a newer Hybrid Access card (TNT-SL-HDLC2 or TNT-SL-HDLC2-EC-C), and if you replace a MAX TNT Series56™ modem card (TNT-SL-48MOD-S56) with a newer Series56 card (TNT-SL-48MOD-S-C or TNT-SL-48MODV3-S-C), you must write new profiles for the new cards.

If you replace a Series56 modem card (TNT-SL-48MOD-S56, TNT-SL-48MOD-SGL, TNT-SL-48MOD-S-C or TNT-SL-48MODV3-S-C) with a MultiDSP card (TNTP-SL-ADI-C, TNTV-SL-ADI-C, or APX8-SL-96DSP), you must write new profiles for the new cards.

For any slot whose card type is being changed, you should perform a `slot -r` command after downing (`slot -d`) or removing the existing card prior to inserting a new card type.

Caveats in this release

- As new features are added to each TAOS release, the amount of memory used by the operating system increases. MAX TNT units will therefore report less available memory with each subsequent release.
- Lucent Technologies does not recommend the use of traffic shaping in TAOS 9.0. (This limitation is removed in TAOS 9.0.2.)
- With TAOS 9.0, when a long-term loss of communication to a RADIUS or call-logging server occurs (which results in loss of data), the MAX TNT unit reports the event by generating a Warning 104 message.

- When you attempt to initiate terminal services such as TCP-clear, Rlogin, or Telnet using a scripted login, the MAX TNT unit might occasionally terminate calls abnormally, displaying a cause code 51 and progress code 40 in Syslog or RADIUS accounting records. This issue is aggravated by scripted logins when the responses are entered before the MAX TNT prompts input.
- In this release, the transmit rate of the Ethernet-3-ND slot card decreases by as much 20% for 60-byte packets when the traffic passing through exceeds its maximum throughput rate.
- Before changing an ATM connection's VPI-VCI assignment, you must disable the connection on a MAX TNT OC3 (Copper) ATM slot card (TNT-SL-OC3-C) or MAX TNT OC3 (Fibre) ATM slot card (TNT-SL-OC3-F).
- Multilink Protocol (MP) bonding of analog calls is supported, but some client modems and software may have compatibility problems.
- Configurable receive and transmit data rate limits are not supported on the MAX TNT unchannelized DS3-ATM slot card (TNT-SL-UDS3A). Configurable receive and transmit data rate limits *are* supported on the unchannelized DS3 Frame slot card (TNT-SL-UDS3).
- LAN-Modem profiles contain entries for 96 devices. For the 96-port MultiDSP card, all 96 entries in the profile are used. For 48-port Digital Modem cards—Series56 (TNT-SL-48MOD-S56), Series56 II (TNT-SL-48MOD-S-C), and Series56 III (TNT-SL-48MODV3-S-C)— only the first 48 entries are used. For the 48-port MultiDSP card (TNTP-SL-ADI-C or TNTV-SL-ADI-C), every other entry in a LAN-Modem profile is used (odd ports only, from 1 to 95).
- Frame Relay and ATM SVCs are not supported in this release.

Built-in features in TAOS 9.0

WAN access server features

Support for the Ethernet-3-ND slot card

The MAX TNT Ethernet-3-ND (no dongle) slot card has four full-duplex 10/100-megabit (Mb) Ethernet ports with a high packet-per-second throughput to support voice over IP (VoIP).

You can view or download the Ethernet 3-ND slot card configuration guide at `http://www.lucent.com/ins/doclibrary/library.html`.

Support for the E3-ATM slot card

The MAX TNT E3-ATM slot card inserts and extracts ATM cells from an E3 stream in full-duplex mode at speeds of up to 34.368 Mbps, for routing applications. Each slot card has its own memory processors and provides connectors for one pair of transmit and receive lines and one backup pair. Two E3-ATM slot cards in the same unit can be connected together and configured for redundancy.

You can view or download the E3-ATM slot card configuration guide at `http://www.lucent.com/ins/doclibrary/library.html`.

Support for the PCTFI slot card

The PCTFI slot card provides a direct connection between the time slot interchanger (TSI) in a 5ESS SM-2000 switch and a MAX TNT unit, eliminating the need for separate T1 or E1 digital trunk interfaces in the switch.

You can view or download the PCTFI slot card configuration guide at You can view or download the Ethernet 3-ND slot card configuration guide at `http://www.lucent.com/ins/doclibrary/library.html`.

Support for port-speed configuration on Ethernet-3 slot cards

This release provides support for 10-Mbps and 100-Mbps port speeds on Ethernet-3 (ENET-3) slot cards. The new `media-speed-mbit` parameter allows you to set either port speed in the Ethernet profile for the card. Prior to this release, the port speed for the Ethernet-3 slot card was limited to 100Mbps.

Parameter	Specifies
`media-speed-mbit`	Speed of the Ethernet port, either 10mb or 100mb. The default is 100Mbps.

In the following example, the administrator sets the speed of port 1 of the Ethernet-3 slot card in slot 8 to 100Mbps:

```
admin> read ethernet { 1 8 1}
ETHERNET/{ shelf-1 slot-8 1 } read
```

```
admin> set media-speed-mbit = 100mb
admin> write
ETHERNET/{ shelf-1 slot-8 1 } written
```

ISDN PRI support for the STM-0 slot card

In this release, TAOS 9.0 provides support for ISDN Primary Rate Interface (PRI) signaling on the Synchronous Transport Module 0 (STM-0) slot card.

The Synchronous Transport Module 0 (STM-0) slot card now supports ISDN Primary Rate Interface (PRI) signaling over T1 lines. ISDN PRI on the STM-0 slot card behaves and is configured similarly to ISDN PRI on other MAX TNT and APX 8000 line cards such as the T3 (Channelized DS3 WAN) slot card and the Peripheral Control Timing Facility Interface (PCTFI) slot card.

As in earlier releases, you configure an STM profile and up to 28 T1 profiles for an STM-0 slot. In the following example, the administrator sets ISDN PRI on the STM-0 slot card in slot 2.

```
admin> read t1 { 1 2 1 }
admin> set line signaling = isdn
admin> set line channel 24 channel = d-channel
admin> set line isdn-emulation-side = nt
admin> set line clock-priority = high
admin> write
```

Traffic shaping on DS3-ATM2 and E3-ATM slot cards

The MAX TNT unit supports ATM traffic shaping only on second-generation ATM slot cards: DS3-ATM2 (APX8-SL-UDS3-A2-C) and E3-ATM (APX8-SL-UE3-A-C). Each ATM interface supports up to 15 traffic shapers that define characteristics for different types of traffic. For example, voice traffic requires a constant amount of bandwidth and cannot tolerate delays, whereas file transfer can tolerate delay and variable bandwidth. Once you have specified the traffic shapers you need, you can apply a shaper to any number of virtual circuits.

Note: MAX TNT units do not support “on the fly” traffic shaper configuration. If you modify a traffic shaper after it has been applied to virtual circuits, you must restart all of the virtual circuits to use the new values.

A MAX TNT unit with DS3-ATM2 or E3-ATM slot cards installed supports the following ATM service categories:

- Constant Bit Rate (CBR)
- Variable Bit Rate-Non-Real-Time (VBR-NRT)
- Unspecified Bit Rate (UBR)

CBR is used for applications that do not tolerate delay (for example, voice or video transmission). It guarantees that a static amount of bandwidth (the maximum effective bit rate) is always available to the circuit. The source system can send cells at or below that bit rate without compromising the quality of service.

VBR-NRT is used for applications such as transaction processing, which can tolerate delay but not cell loss. The bandwidth must remain within the boundaries of the maximum effective bit rate, the maximum sustainable bit rate, and the maximum burst size.

UBR is the lowest level of service. In effect, it makes no service or bandwidth guarantees and does not enforce traffic management. UBR is used for applications such as telecommuting or background data transfer, which can tolerate delay.

For a detailed definition of the ATM service categories, see the *ATM Forum Traffic Management Specification Version 4.0*.

Overview of traffic-shaping settings

The following parameters (shown with default values) define traffic shaping on DS3-ATM2 and E3-ATM slot cards:

```
[in DS3-ATM/{ any-shelf any-slot 0 }:line-config:traffic-shapers[1]]
enabled = no
bit-rate = 1000
peak-rate = 1000
max-burst-size = 2
aggregate = no
priority = 0
```

```
[in E3-ATM/{ any-shelf any-slot 0 }:line-config:traffic-shapers[1]]
enabled = no
bit-rate = 1000
peak-rate = 1000
max-burst-size = 2
aggregate = no
priority = 0
```

```
[in CONNECTION/"":session-options]
traffic-shaper = 16
```

Parameter	Specifies
Enabled	Enable/disable the shaper for use.
Bit-Rate	Maximum sustainable effective bit rate in kilobits per second. The default is `1000` (1Mbps). This setting applies only to VBR traffic. For more information about this setting, see “Framing and effective line rates” on page 17.
Peak-Rate	Maximum effective bit rate allowed, in kilobits per second. The default is `1000` (1Mbps). For CBR traffic, this setting specifies the static bit rate. For VBR traffic, it is the upper boundary of the variable bit rate. For more information about this setting, see “Framing and effective line rates” on page 17.
Max-Burst-Size	Maximum burst size (MBS), which is the maximum number of cells that can be transmitted at the specified peak rate before the MAX TNT unit determines that the virtual circuit is exceeding the defined characteristics. The default is `2`. The valid range is from 2 through 255. This setting applies only to VBR traffic.

Parameter	Specifies
Aggregate	*Not supported on E3-ATM and DS3-ATM2 slot cards.* If a traffic shaper with this parameter set to `yes` is applied to more than one virtual circuit, each of the virtual circuits will be allowed the full bandwidth defined in the shaper. For example, if the shaper specifies CBR service and a peak rate of 10000, and two virtual circuits apply the shaper, each circuit will be allowed 10 Mbps of bandwidth, for a total of 20 Mbps. **Note:** MAX TNT does not verify that the aggregate rates of the multiple virtual circuits do not exceed the effective line rate.
Priority	ATM service category. The default value, 0 (zero), specifies CBR service. Other supported values are 1, for VBR-NRT service, and 2, for UBR service.
Traffic-Shaper	Number of a defined traffic shaper to apply to the virtual circuit. The default is shaper `16`, which is an internal shaper that is not configurable. Traffic shaper 16 specifies no bandwidth limitation.

Transmit resource sharing

The E3-ATM and DS3-ATM2 transmit buffers are separated into eight pools. The first seven pools are dedicated to traffic shapers 1 through 7. The remaining pool is shared among the remaining traffic shapers (shapers 8 through 15). Note that when assigning a traffic shaper to multiple virtual circuits, if multiple virtual circuits share the same pool, the circuits compete for pool resources.

Note: The MAX TNT unit uses a special pool for OAM traffic to ensure that OAM cells are not lost because of congestion. OAM cells are handled with priority 0.

Using traffic shapers 1 through 7

The first seven traffic shapers each have a dedicated pool. When you apply a traffic shaper from 1 through 7 to two virtual circuits, each virtual circuit applies the same settings and calculates the same transmit bandwidth limitation. For example, suppose traffic shaper 1 has the following settings:

```
[in DS3-ATM/{ shelf-1 trunk-module-1 1 }:line-config:traffic-shapers[1]]
enabled = yes
bit-rate = 1000
peak-rate = 15000
max-burst-size = 2
aggregate = no
priority = 0
```

And suppose that this traffic shaper is applied to two virtual circuits, as follows:

```
admin> read connection atmvc-1
CONNECTION/atmvc-1 read

admin> set session-options traffic-shaper = 1

admin> write
CONNECTION/atmvc-1 written

admin> read connection atmvc-2
CONNECTION/atmvc-1 read
```

```
admin> set session-options traffic-shaper = 1
admin> write
CONNECTION/atmvc-2 written
```

When traffic shaper 1 is applied to two virtual circuits, each virtual circuit has a maximum effective bit rate of 15 Mbps. The bandwidth limitation works as expected when the traffic for each virtual circuit is below the bandwidth limit. However, if the traffic on one of the virtual circuits exceeds its bandwidth limit, that circuit can consume all of the transmit buffers in the pool, which would deprive the other virtual circuit of transmit buffers and prevent it from reaching its bandwidth limit. The virtual circuit with the highest traffic has a statistically higher chance of obtaining pool resources. In the case where both virtual circuits exceed their limit by the same amount, they compete equally for resources, enabling each to attain its limit.

Using traffic shapers 8 through 15

When two virtual circuits are governed by different shapers using the common pool (for example, traffic shapers 8 and 9), the situation is further complicated by the fact that each virtual circuit can have a different bandwidth limit. For example, if traffic shaper 8 specifies 1Mbps and traffic shaper 9 specifies 7Mbps, the virtual circuits that use these shapers compete for the transmit buffers in the common pool used by traffic shapers 8 through 15.

Framing and effective line rates

MAX TNT checks that the `peak-rate` and `bit-rate` values of a specific shaper do not exceed the effective line rate. However, the aggregate rates of the virtual circuits using the shaper are not checked.

Traffic shapers are precise to within 4%.

Table 13 shows the effective line rates for DS3-ATM2 and E3-ATM slot cards with the supported framing formats. Effective line rate indicates the usable bandwidth after framing and ATM cell headers have been taken into account.

Table 13. Framing and effective line rate

Slot card	Framing	Effective line rate	Cells per second
DS3-ATM2	C-bit-ADM	40,037 Kbps	104,265
	C-bit-PLCP	36,864 Kbps	96,000
E3-ATM	G751-ADM	30,474 Kbps	79,360
	G751-PLCP	27,648 Kbps	72,000
	G832-ADM	30,720 Kbps	80,000

The minimum shaper value is 1Kbps. If the `peak-rate` or `bit-rate` parameters specifies value that is less than the minimum shaper value, the system logs a warning message.

Disabled traffic shapers

If a shaper is disabled and a virtual circuit has that shaper applied, or when the shaper #16 is used by a Connection profile, the virtual circuit has a CBR QoS with a peak rate equal to the maximum line bandwidth. This virtual circuit shares the common transmit pool.

Firmware versions for digital modems

The Conexant firmware versions for MAX TNT Digital Modem cards include support for V.90, K56flex, K56plus, and all slower, standard modem speeds. This release includes the following Conexant firmware:

- Series56 Digital Modem cards (also called CSM/1, TNT-SL-48MOD-S56) support V2.098-K56_2M_DLP_CSM firmware.
- Series56 II Digital Modem cards (also called CSM/3, TNT-SL-48MOD-SGL and TNT-SL-48MOD-S-C) support V5.817 firmware.
- Series56 III Digital Modem cards (also called CSMV/3, TNT-SL-48MODV3-S-C) support V5.8175 firmware.

Firmware versions for MultiDSP cards

This release includes the following Lucent firmware versions for MultiDSP cards:

- 48-port MultiDSP cards (TNTP-SL-ADI-C or TNTV-SL-ADI-C) support Lucent firmware.
- 96-port MultiDSP cards (APX8-SL-96DSP) support Lucent V0.1622.0 firmware.

Support for HDLC-NRM on Series56 III Digital Modem cards

In this release, the Series56 III Digital Modem slot card (also known as the CSMV/3 card) supports analog synchronous connections that use the High-Level Data Link Control-Normal Response Mode (HDLC-NRM) protocol. You configure the connections just as you would for a Series56 II Digital Modem card (also known as an CSM3 card). For details, see the *APX 8000/MAX TNT/DSLTNT WAN, Routing, and Tunneling Configuration Guide* for TAOS 8.0.

Support for V.110 subrates for MultiDSP cards

This release includes new 14400-bps and 28800-bps V.110 subrates for 48-port and 96-port MultiDSP slot cards.

A total of eight new switched-call types have been added for V.110 connections. They indicate the bearer channel capability to set up for each switched call in the session. Two existing parameters have these new values:

- Data-Service in the Connection > Telco-Options subprofile
- Switched-Call-Type in the Frame-Relay profile

Each value specifies that the call be set up for either 56-Kbps or 64-Kbps data transfer that is either restricted or nonrestricted, and a V.110 bit rate of either 14400 bps or 28800 bps.

Value	Call type requested
`144-56k-v110`	56Kbps unrestricted data transfer and a V.110 bit rate of 14400-bps.
`288-56k-v110`	56Kbps unrestricted data transfer and a V.110 bit rate of 28800-bps.
`144-56kr-v110`	56Kbps restricted data transfer and a V.110 bit rate of 14400-bps.
`288-56kr-v110`	56Kbps restricted data transfer and a V.110 bit rate of 28800-bps.
`144-64k-v110`	64Kbps unrestricted data transfer and a V.110 bit rate of 14400-bps.
`288-64k-v110`	64Kbps unrestricted data transfer and a V.110 bit rate of 28800-bps.
`144-64kr-v110`	64Kbps restricted data transfer and a V.110 bit rate of 14400-bps.
`288-64kr-v110`	64Kbps restricted data transfer and a V.110 bit rate of 28800-bps.

Note: If a V.110 device makes a call at 14400 bps or 28800 bps to a MAX TNT with a MultiDSP or MultiDSP2 card, the call automatically connects at 14400 bps or 28800 bps, regardless of the setting of Data-Service or Switched-Call-Type.

Support for progress code 33

A new progress code, 33 (`modem awaiting remote modem`), indicates that a modem has failed to synchronize because it did not detect a remote analog client modem. Progress Code 33 is supported only in the 48-port and 96-port MultiDSP slot cards.

In previous releases, this condition was indicated by progress code 31, which was returned when a user dialed into a MAX TNT modem and failed to synchronize for one of three possible reasons.

Progress code 33 indicates that the call did not originate from a remote client modem. When a modem is opened and is waiting for the carrier, the accounting record is updated with progress code 33. If the remote modem client is detected, the accounting record is updated to progress code 31.

You can view progress code 33 from the console log, or Syslog, or by means of SNMP queries.

Authentication and accounting client features

Support for CHAP name challenge during incoming calls

You can now set a Challenge Handshake Authentication Protocol (CHAP) challenge name for bidirectional CHAP authentication of incoming calls. The new Substitute-Send-Name parameter in the PPP-Answer subprofile of the Connection profile provides a unique, substitute name for the calling host to which the MAX TNT connects during incoming calls.

Because bidirectional CHAP authentication provides a way to formally authenticate the calling device during an incoming call, the name of the device must be checked against a locally defined name. The name can be the dial-in profile name or the substituted name provided by this new parameter.

Although you set this parameter in the PPP-Answer subprofile, the PPP-Options subprofile in the Connection profile includes a copy of this setting.

Following are the relevant settings for configuring the unit to use the CHAP challenge name:

```
[in ANSWER-DEFAULTS:ppp-answer]
admin> set substitute-send-name = groupb
admin> write
ANSWER-DEFAULTS written
```

Parameter	Specifies
`substitute-send-name`	Name of the PPP calling device during incoming calls to the MAX TNT unit, a name of up to 23 characters. The default is a null string. If no value is entered, the global system name is used.

Nonauthentication option for asynchronous framed users

You can now turn user authentication off for all incoming calls from asynchronous framed users, while continuing to restrict access to other types of users. When the feature is enabled, a MAX TNT unit requires no user authentication during Link Control Protocol (LCP) negotiation. Instead, users are automatically assigned to an IP address pool set aside for their sole use. In addition, a user who fails Password Authentication Protocol (PAP) authentication for the network connection can attempt the connection again up to five times. To implement this feature, you must set two new parameters in the PPP-Answer subprofile and one new parameter in the IP-Answer subprofile. Both subprofiles are in the Answer-Defaults profile.

Auth-for-Async-Framed-User

Description: Enables and disables the authentication requirement for incoming asynchronous framed users.

Usage: Specify Required or Not-Required. The default is Required.

- Required enables the authentication requirement for incoming asynchronous framed users.
- Not-Required disables the authentication requirement. These users without authentication are automatically assigned to an IP address pool set aside for their use.

Example: `set auth-for-async-framed-user = not-required`

Dependencies: Consider the following:

- If this parameter is set to Not-Required, you must assign a pool number with the Pool-for-Async-Framed-User parameter to provide IP addresses for incoming asynchronous framed users without authentication.
- A read-only copy of this parameter appears in the IP-Options subprofile.
- Consider allowing users to retry PAP authentication after failures by setting the Max-Pap-Auth-Retry parameter.
- **Location:** Answer-Defaults > PPP-Answer, Connection > PPP-Options

Max-Pap-Auth-Retry

Description: Determines the maximum number of retries allowed if PAP authentication for network connection fails.

Usage: Specify a number between 0 and 5. The default is 0 raters.

Example: `set max-pap-retry = 0`

Dependencies: A read-only copy of this parameter appears in the IP-Options subprofile.

Location: Answer-Defaults > PPP-Answer, Connection > PPP-Options

Pool-for-Async-Framed-User

Description: Assigns an IP address pool of a particular number for incoming asynchronous framed users without authentication.

Usage: Specify an IP pool number between 0 and 512. The default is 0, which allows the unit to assign an address from any pool.

Example: `set pool-for-async-framed-user = 0`

Dependencies: Consider the following:

- If the Auth-for-Async-Framed-User parameter is set to Not-Required, you must assign a pool number to provide IP addresses for incoming asynchronous framed users without authentication.
- Because this pool is for the sole use of asynchronous framed users without authentication, the MAX TNT cannot allocate an IP address from this same pool to incoming users *with* authentication.
- A read-only copy of this parameter appears in the PPP-Options subprofile.

Permission level: Answer-Defaults > IP-Answer, Connection > IP-Options

Service-Type (6) set to Call-Check for CLID and DNIS authentication

When using CLID or DNIS authentication, MAX TNT units that are operating in vendor-specific attribute (VSA) compatibility mode now generate a RADIUS authentication request that includes Call-Check (10) as the Service-Type value. If the unit is not operating in VSA-compatibility mode, the Service-Type attribute retains the old Outbound value.

Following are the relevant settings for configuring the unit in VSA-compatibility mode:

```
[in EXTERNAL-AUTH]
auth-type = radius
```

```
[in EXTERNAL-AUTH:rad-auth-client]
auth-radius-compat = vendor-specific
```

Following is an example RADIUS log from a MAX TNT unit operating in VSA compatibility mode, with the Call-Check value shown:

```
May 2 13:50:47.429 radiusd[23414] handle_radius_request:
   1.1.1.1:1025, id=18, code=1, length=104
   request: User-Name = "63210"
   request: User-Password = "\x01{6\xe7\xf2\xc0g\xa5\xbb\x02\x05@"
   request: NAS-IP-Address = 192.168.21.210
```

```
request: NAS-Port = 10501
request: NAS-Port-Type = ISDN-Sync
request: Service-Type = Call-Check
request: State = ""
request: Calling-Station-Id = "63210"
request: Called-Station-Id = "63222"
request: Acct-Session-Id = "2161100"
request: Ascend-Data-Rate = 56000
request: Ascend-Xmit-Rate = 56000
```

Service-Type attribute (6) in RADIUS accounting records

For connections authenticated using RADIUS, if the user profile returned by the RADIUS server includes a Service-Type (6) attribute, the RADIUS Start and Stop accounting records will now report the Service-Type value. The following conditions must be met for the Service-Type value to appear in accounting records:

- External authentication must be enabled and configured as RADIUS. Note that RADIUS/LOGOUT will not exhibit this feature.
- RADIUS accounting must be enabled.
- RADIUS accounting compatibility must be set to `vendor-specific` via the Acct-RADIUS-Compat parameter in the TAOS command-line interface.
- The RADIUS user profile must contain a Service-Type return attribute, and the connection must be authenticated by means of the RADIUS profile. (A connection authenticated by means of a local profile does not report the Service-Type value.)

In addition to being included in Access-Request and Access-Accept packets, the RADIUS Service-Type attribute now appears in RADIUS accounting records.

Following is an example of a RADIUS Start Record, with a sample Service-Type value:

```
User-Name = "jimtest"
NAS-IP-Address = 212.168.21.90
NAS-Port = 2054
NAS-Port-Type = Sync
Service-Type = Framed
Class = "testclass"
Acct-Status-Type = Start
Acct-Delay-Time = 0
Acct-Session-Id = "317839070"
Acct-Authentic = RADIUS
Ascend-Multilink-ID = 358219783
Ascend-Num-In-Multilink = 0
Ascend-Modem-PortNo = 3
Ascend-Modem-SlotNo = 2
Ascend-Modem-ShelfNo = 1
Framed-Protocol = MPP
Framed-IP-Address = 212.168.21.93
```

RADIUS: New authentication-delay attribute

The new Ascend-Auth-Delay attribute has been added to the RADIUS dictionary, for call-logging and RADIUS accounting purposes.

Ascend-Auth-Delay (28)

Description: Indicates the amount of time (in milliseconds) in which the system carried out the authentication process.

Usage: The Ascend-Auth-Delay attribute appears in call-logging or RADIUS accounting Start packets, Stop packets, or Checkpoint packets.

Example: `ascend-auth-delay = 20`

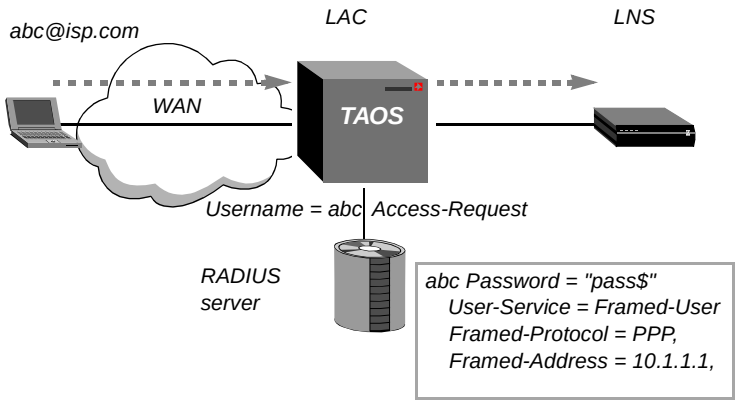
See Also: Ascend-Auth-Type

Stripping portions of the username from RADIUS access requests

MAX TNT units are now able to strip off portions of the username sent in the Username attribute-value pair of a RADIUS Access-Request packet. The primary purpose of this feature is to remove the domain name from incoming authentication requests. However, the feature is not limited to that purpose.

Figure 1 shows a MAX TNT unit that is functioning as a Layer 2 Tunneling Protocol (L2TP) access concentrator (LAC). The unit is removing the portion of the username value that follows the *at* sign (@) before forwarding the Username attribute-value pair in a RADIUS Access-Request packet.

Figure 1. Removal of domain name before RADIUS authentication



The user dialing into the MAX TNT unit has the following username:

`abc@isp.com`

On the RADIUS server, the user profile name is `abc`. To enable the authentication to take place properly, the MAX TNT unit must remove `@isp.com` before forwarding the Access-Request packet to the RADIUS server. After the client has been authenticated by the MAX TNT unit operating as a LAC, the LAC forwards the username and password to the L2TP network server (LNS).

Overview of parameter settings

To provide flexibility in processing Username values to be forwarded to a RADIUS server in Access-Request packets, you can specify one or multiple characters as delimiters, the number of delimiters that must be present in a username for the unit to strip off characters, and whether to strip characters to the left or right side of the specified delimiter characters.

Following are the parameters, shown with default values, for specifying the delimiters and the direction in which the Username value will be modified:

```
[EXTERNAL-AUTH:rad-auth-client]
auth-realm-delimiters = /\@%
auth-req-delim-count = 0
auth-req-strip-side = none
```

Parameter	Specifies
Auth-Realm-Delimiters	Character or characters to be recognized as delimiters in a username. In previous releases, the delimiters specified by this parameter were applied to Access-Accept packets only, and were used to define realms. With the current software, the delimiters are also used to define the boundaries of characters to be stripped from the username in Access-Request packets. The default value `/\@%` consists of the characters typically used for delimiting realms and domain names. You can specify up to 7 characters in any order.
Auth-Req-Delim-Count	Number of delimiter characters to delete. With the default zero value, no characters are stripped from the name. If the number of delimiters in the username is greater than or equal to the value of this parameter, the unit strips the characters to the left or right (as specified in the Auth-Req-Strip-Side setting) and sends the remaining string in the Username attribute-value pair. If the number of delimiters in the username is *less than* the value of the Auth-Req-Delim-Count parameter, the unit sends the entire username to RADIUS without stripping any characters.
Auth-Req-Strip-Side	Direction in which to strip characters from a username. The default value is `none`, which specifies that the unit removes no characters before sending the Username attribute-value pair. Other valid values are `left` (strip the delimiter character and characters to the left of it) and `right` (strip the delimiter character and characters to the right of it).

Example of configuring a MAX TNT unit to remove domain names

In this example, a user logs in to a MAX TNT unit with the following username:

```
billg@abc.com%xzy^msn.com
```

Following is the user's RADIUS profile:

```
billg Password = "localpw"
    User-Service = Framed-User,
    Framed-Protocol = PPP,
    Framed-Address = 1.2.3.4,
    Framed-Netmask = 255.255.255.255
```

The following commands configure a MAX TNT unit to remove the *at* sign (@) and all characters to the right of it in the name the user presents at login:

```
admin> read external-auth
EXTERNAL-AUTH read

admin> set rad-auth-client auth-realm-delimiters = @
```

```
admin> set rad-auth-client auth-req-delim-count = 1

admin> set rad-auth-client auth-req-strip-side = right

admin> write
EXTERNAL-AUTH written
```

With this configuration, when the MAX TNT unit receives a RADIUS-authenticated call from a user with the following name, the unit strips the delimiter character and all characters to the right of it and sends the remaining string (`billg`) in the Username attribute-value pair:

```
billg@abc.com%xzy^msn.com
```

Example of configuring a MAX TNT unit to recognize various delimiters

In this example, three users log in to a MAX TNT unit with the following usernames:

```
abc\isp2.com

def@isp3.com

hij/isp4.com
```

Following are the users' RADIUS profiles:

```
abc Password = "localpw"
    User-Service = Framed-User,
    Framed-Protocol = PPP,
    Framed-Address = 1.2.3.4,
    Framed-Netmask = 255.255.255.255

def Password = "localpw"
    User-Service = Framed-User,
    Framed-Protocol = PPP,
    Framed-Address = 2.3.4.5,
    Framed-Netmask = 255.255.255.255

hij Password = "localpw"
    User-Service = Framed-User,
    Framed-Protocol = PPP,
    Framed-Address = 3.4.5.6,
    Framed-Netmask = 255.255.255.255
```

The following commands configure a MAX TNT unit to remove all characters to the right of one of the specified delimiters in the name the user presents at login:

```
admin> read external-auth
EXTERNAL-AUTH read

admin> set rad-auth-client auth-realm-delimiters = /\@%

admin> set rad-auth-client auth-req-delim-count = 1

admin> set rad-auth-client auth-req-strip-side = right

admin> write
EXTERNAL-AUTH written
```

With this configuration, when the MAX TNT unit receives a RADIUS-authenticated call from a user with one of the following names the unit strips the delimiter character and all characters to the right of it and sends the remaining string (`abc`, `def`, or `hij`) in the Username attribute-value pair:

```
abc\isp2.com
```

```
def@isp3.com
hij/isp4.com
```

Example of configuring a MAX TNT to require multiple delimiters in a name

In this example, two callers log in to a MAX TNT unit with the following usernames:

```
abc@def@isp1.com
ghi@jkl%isp2.com
```

Following are the users' RADIUS profiles:

```
abc Password = "localpw"
    User-Service = Framed-User,
    Framed-Protocol = PPP,
    Framed-Address = 1.2.3.1,
    Framed-Netmask = 255.255.255.0

ghi Password = "localpw"
    User-Service =Framed-User
    Framed-Protocol = PPP,
    Framed-Address = 2.3.4.5,
    Framed-Netmask = 255.255.255.248
```

The following commands configure a MAX TNT unit to remove all characters to the right of the first (leftmost) delimiter if the name contains two or more delimiters:

```
admin> read external-auth
EXTERNAL-AUTH read

admin> set rad-auth-client auth-realm-delimiters = /@%\

admin> set rad-auth-client auth-req-delim-count = 2

admin> set rad-auth-client auth-req-strip-side = right

admin> write
EXTERNAL-AUTH written
```

With this configuration, when the MAX TNT unit receives a RADIUS-authenticated call from a user with one of the following names, the unit removes the first delimiter character and all characters to the right of it (including the second delimiter character and its following text). The unit then sends the remaining string (`abc` or `ghi`) in the Username attribute-value pair:

```
abc@def@isp1.com
ghi@jkl%isp2.com
```

If a user dials in with the following name, the call fails:

```
abc@isp1.com
```

When the unit determines that the name contains fewer than the specified number of delimiters, it passes the name to the RADIUS server without stripping any characters.

CLID and DNIS authentication cause codes

A new `bt-ss7` switch type has been introduced for use on Signaling System 7 (SS7) data trunks. The new switch type is equivalent to the `net5-pri` setting, except for the codes

returned for calling line ID (CLID) or the Dialed Number Information Service (DNIS) authentication failure.

The purpose of the new setting is to enable MAX TNT units to return the Q.850 disconnect cause code 63 (`service not available`) and location 10 (`network beyond interworking point`) if a call is rejected because of CLID or DNIS authentication failure. This behavior is consistent with British Telecom call processing recommendations for SS7-IP interworking.

Following are the relevant parameters, shown with default values:

```
in T1/{ any-shelf any-slot 0 }:line-interface]
signaling-mode = inband
switch-type = att-pri
```

```
[in E1/{ any-shelf any-slot 0 }:line-interface]
signaling-mode = inband
switch-type = net5-pri
```

Parameter	Specifies
Signaling-Mode	Type of signaling used on a T1 or E1 line. For the MAX TNT unit to use this feature, the parameter must be set to `ss7-data-trunk`.
Switch-Type	Type of network switch. The new `bt-ss7` setting is equivalent to the `net5-pri` setting except for the cause codes returned for CLID or DNIS authentication failure. With the `net5-pri` or any switch-type setting other than `bt-ss7`, if a call is rejected because of CLID or DNIS authentication failure, the MAX TNT unit releases the call with cause code 16 (`normal clearing`) and location 0 (`user`). If the parameter is set to `bt-ss7`, the MAX TNT unit releases the call with cause code 63 (`service not available`) and location 10 (`s`). The setting takes effect as soon as the profile is written.

For example, the following commands enable this feature on an E1 line in shelf 1, slot 7:

```
admin> read e1 { 1 7 1}
E1/{ shelf-1 slot-7 1 } read
```

```
admin> set line-interface signaling-mode = ss7-data-trunk
```

```
admin> set line-interface switch-type = bt-ss7
```

```
admin> write
E1/{ shelf-1 slot-7 1 } written
```

If you attempt to save the profile with the `bt-ss7` switch type and any other Signaling-Mode setting, the system displays the following error message:

```
Switch type is not appropriate for the configured signaling type.
```

16-bit vendor-specific attribute (VSA) support

In previous releases, the VSA type was encoded in an 8-bit field, which limits RADIUS and call-logging configurations to a maximum of 256 attribute types. In this release, the VSA type is encoded in a 16-bit field, which substantially increases the number of attribute types available. Note that only the NavisRadius™ product supports 16-bit VSAs at this time.

VSA formats

VSA support now accommodates three formats: standard RFC, 8-bit VSA, and 16-bit VSA.

Standard RFC format

All standard RFC 2058 attributes use the following format.

```
0                   1                   2
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Attr Type     |   Length      |  Value ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

If Attr-Type is not Vendor-Specific, the system uses the standard RFC format to decode the attribute.

8-bit VSA format

Following is the 8-bit VSA format:

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Attr Type     |   Length      |           Vendor-Id
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Vendor-Id (cont)          | Vendor Type(8)|  Vendor length|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Vendor-value......
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

When you use the 8-bit VSA format, Attr-Type is set to Vendor-Specific (26) and Vendor-Id is set to Ascend-Vendor-Id (529).

16-bit VSA format

Following is the 16-bit VSA format:

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Attr Type     |   Length      |           Vendor-Id
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Vendor-Id (cont)          |         Vendor Type(16-bit)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Vendor Length |  Vendor-value......
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

Consider the following:

- Attr-Type is set to Vendor-Specific (26).
- Vendor-Id is set to Lucent-Vendor-Id (4846).
- Vendor Length is set to the octet count of the Vendor Type, Vendor Length and Vendor-value.

Note: At this time, only NavisRadius supports 16-bit VSAs.

User interface changes

The following parameters have been modified:

- Acct-RADIUS-Compat
- Auth-RADIUS-Compat
- Call-Log-RADIUS-Compat
- RADIUS-Server-Compat

Descriptions of each modified parameter appear below.

Acct-RADIUS-Compat

Description: Enables or disables vendor-specific attribute (VSA) compatibility mode when the unit is using Remote Authentication Dial-In User Service (RADIUS) for accounting purposes.

Usage: Specify one of the following settings:

- Old-Ascend (the default) specifies that the unit does not send the Vendor-Specific attribute to the RADIUS server and does not recognize the Vendor-Specific attribute if the server sends it. All attributes are sent in standard RFC format.
- Vendor-Specific specifies 8-bit VSA support. All standard attributes are sent in standard RFC format, and all VSAs are sent in 8-bit VSA format. The unit ignores all VSAs in received packets that do not have Vendor-Id set to Ascend-Vendor-Id.
- 16-Bit-Vendor-Specific specifies 16-bit VSA support. All standard attributes are sent in standard RFC format, and all VSAs are sent in the 16-bit VSA format as Lucent VSAs. The system ignores all VSAs in received packets that do not have Vendor-Id set to Lucent-Vendor-Id. In this format, the first 256 Lucent VSAs are mapped to the 256 Ascend VSAs.

  Note: At this time, only NavisRadius supports 16-bit VSAs.

Example: `set acct-radius-compat = vendor-specific`

Location: External-Auth > Rad-Acct-Client

See Also: Auth-RADIUS-Compat, Call-Log-RADIUS-Compat, RADIUS-Server-Compat

Auth-RADIUS-Compat

Description: Enables or disables vendor-specific attribute (VSA) compatibility mode when the unit is using Remote Authentication Dial-In User Service (RADIUS) for authentication and authorization purposes.

Usage: Specify one of the following settings:

- Old-Ascend (the default) specifies that the unit does not send the Vendor-Specific attribute to the RADIUS server and does not recognize the Vendor-Specific attribute if the server sends it. All attributes are sent in standard RFC format.
- Vendor-Specific specifies 8-bit VSA support. All standard attributes are sent in standard RFC format, and all VSAs are sent in 8-bit VSA format. The unit ignores all VSAs in received packets that do not have Vendor-Id set to Ascend-Vendor-Id.
- 16-Bit-Vendor-Specific specifies 16-bit VSA support. All standard attributes are sent in standard RFC format, and all VSAs are sent in the 16-bit VSA format as Lucent VSAs. The system ignores all VSAs in received packets that do not have Vendor-Id set to Lucent-Vendor-Id. In this format, the first 256 Lucent VSAs are mapped to the 256 Ascend VSAs.

  Note: At this time, only NavisRadius supports 16-bit VSAs.

Example: `set auth-radius-compat = vendor-specific`

Location: External-Auth > Rad-Auth-Client

See Also: Acct-RADIUS-Compat, Call-Log-RADIUS-Compat, RADIUS-Server-Compat

Call-Log-RADIUS-Compat

Description: Enables or disables vendor-specific attribute (VSA) compatibility mode when the unit is using Remote Authentication Dial-In User Service (RADIUS) for call logging to the NavisAccess™ manager.

Usage: Specify one of the following settings:

- Vendor-Specific specifies 8-bit VSA support. All standard attributes are sent in standard RFC format, and all VSAs are sent in 8-bit VSA format. The unit ignores all VSAs in received packets that do not have Vendor-Id set to Ascend-Vendor-Id.
- 16-Bit-Vendor-Specific specifies 16-bit VSA support. All standard attributes are sent in standard RFC format, and all VSAs are sent in the 16-bit VSA format as Lucent VSAs. The system ignores all VSAs in received packets that do not have Vendor-Id set to Lucent-Vendor-Id. In this format, the first 256 Lucent VSAs are mapped to the 256 Ascend VSAs.

  Note: At this time, only NavisRadius supports 16-bit VSAs.

Example: `set call-log-radius-compat = vendor-specific`

Dependencies: Note that the Old-Ascend setting is no longer available for Call-Log-RADIUS-Compat.

Location: Call-Logging

See Also: Acct-RADIUS-Compat, Auth-RADIUS-Compat, RADIUS-Server-Compat

RADIUS-Server-Compat

Description: Enables or disables Vendor-Specific Attribute (VSA) compatibility mode when the unit is acting as a Remote Authentication Dial-In User Service (RADIUS) server that is able to accept requests for certain limited purposes, such as changing a filter or disconnecting a user.

Usage: Specify one of the following settings:

- Old-Ascend (the default) specifies that the unit does not send the Vendor-Specific attribute to the RADIUS server and does not recognize the Vendor-Specific attribute if the server sends it. All attributes are sent in standard RFC format.
- Vendor-Specific specifies 8-bit VSA support. All standard attributes are sent in standard RFC format, and all VSAs are sent in 8-bit VSA format. The unit ignores all VSAs in received packets that do not have Vendor-Id set to Ascend-Vendor-Id.
- 16-Bit-Vendor-Specific specifies 16-bit VSA support. All standard attributes are sent in standard RFC format, and all VSAs are sent in the 16-bit VSA format as Lucent VSAs. The system ignores all VSAs in received packets that do not have Vendor-Id set to Lucent-Vendor-Id. In this format, the first 256 Lucent VSAs are mapped to the 256 Ascend VSAs.

  Note: At this time, only NavisRadius supports 16-bit VSAs.

Example: `set radius-server-compat = vendor-specific`

Location: External-Auth > Rad-Auth-Server

See Also: Acct-RADIUS-Compat, Auth-RADIUS-Compat, Call-Log-RADIUS-Compat

SNMP changes

Following are the SNMP changes to `call_log.mib`:

- The new `rad16BitVendorSpecific(5)` option has been added to the `callLoggingRadCompatMode` variable.
- The `radOldAscend(3)` option of the `callLoggingRadCompatMode` variable has been deprecated. Set requests using `radOldAscend(3)` return a `BadValue` error message.

Contents of the user-user IE now available to RADIUS

In this release, when the system finds an ISDN user-user information element (IE) in the Setup message for an incoming call, it forwards the information to RADIUS as the value of the new attribute Ascend-UU-Info. The Ascend-UU-Info attribute appears in Access-Request, Accounting-Request, and Checkpoint packets. The user-user information element is forwarded to RADIUS only when the Setup message contains it.

In previous releases, the contents of the ISDN user-user information element in the Setup message for an incoming call were not forwarded to RADIUS. Now, the unit forwards this information to the RADIUS server as the value of the Ascend-UU-Info attribute. Japanese cellular and PHS operators use the user-user information element to determine where the mobile telephone terminal is located. The called party or service provider can use this information to develop application services, such as a restaurant guide in which the caller is located through a Web page.

Attribute description

Ascend-UU-Info (7)

Description: Indicates the contents of the ISDN user-user information element in the Setup message for the incoming call.

Usage: The Ascend-UU-Info attribute appears in Access-Request, Accounting-Request, and Checkpoint packets. The maximum amount of information assigned to the attribute is 240 bytes, and the data is encoded in hexadecimal format.

Following is an Accounting Start record that contains the Ascend-UU-Info attribute:

```
Mon Apr 10 00:46:12 2000
  User-Name = "shobhapipe"
  NAS-IP-Address = 110.110.110.19
  Vendor-Specific = vAscend-560600000000
  NAS-Port = 11
  Vendor-Specific = vAscend-0d0600000002
  NAS-Port-Type = Sync
  Service-Type = Framed
  Acct-Status-Type = Start
  Calling-Station-Id = "53057"
  Ascend-Calling-Subaddress = "6566"
  Ascend-UU-Info = "AABBCCDD"
```

Following is an Accounting Stop record that contains the Ascend-UU-Info attribute:

```
Mon Apr 10 00:46:31 2000
  User-Name = "shobhapipe"
  NAS-IP-Address = 110.110.110.19
  Vendor-Specific = vAscend-560600000000
  NAS-Port = 11
  Vendor-Specific = vAscend-0d0600000002
  NAS-Port-Type = Sync
  Service-Type = Framed
  Acct-Status-Type = Stop
  Calling-Station-Id = "53057"
  Ascend-Calling-Subaddress = "6566"
  Ascend-UU-Info = "AABBCCDD"
```

Dependencies: Ascend-UU-Info is a Vendor-Specific Attribute (VSA).

Local profile information

If the contents of the user-user information element are available in the Setup message, two new parameters, User-User-Info and User-User-Infolen, in the Call-Info profile, display the information and its size in bytes, respectively.

User-User-Info

Description: Indicates the contents of the ISDN user-user information element in the Setup message for the incoming call.

Usage: The User-User-Info value is a read-only hexadecimal value.

Example: `user-user-info=00:04:05:06:07:08:09:10:01:02:03:04:05:+`

Location: Call-Info {*index*}

See Also: User-User-Infolen

User-User-Infolen

Description: Indicates the size (in bytes) of the value of User-User-Info.

Usage: The User-User-Infolen value is read only.

Example: `user-user-infolen = 120`

Location: Call-Info {*index*}

See Also: User-User-Info

Management agent features

SNMP: Increased number of SNMP managers

In previous releases, you could specify up to five Simple Network Management Protocol (SNMP) managers with Read or Write permission. In this release, you can specify up to eight SNMP managers.

Following are the updated descriptions for the Read-Access-Hosts and Write-Access-Hosts parameters in the SNMP profile.

Read-Access-Hosts

Description: An array containing up to eight IP addresses of SNMP managers that have Read permission. If Enforce-Address-Security is set to Yes, the MAX TNT unit responds to SNMP Get and Get-Next commands only from the SNMP managers you specify in the array.

Usage: Each element in the array can specify an IP address. When SNMP is the working profile, you can use the List command to display the array elements. You can then set a value in the Read-Access-Hosts parameter by specifying the numeric index of one of the array elements and the value for that element. Or, you can set an array element without listing the array.

Example: To list the array elements and set the Read-Access-Hosts [1] value:

```
admin> list read-access-hosts
[in SNMP:read-access-hosts]
read-access-hosts[1] = 0.0.0.0
read-access-hosts[2] = 0.0.0.0
read-access-hosts[3] = 0.0.0.0
```

```
read-access-hosts[4] = 0.0.0.0
read-access-hosts[5] = 0.0.0.0
read-access-hosts[6] = 0.0.0.0
read-access-hosts[7] = 0.0.0.0
read-access-hosts[8] = 0.0.0.0
admin> set 1 10.2.3.4/24
```

To set the Read-Access Hosts [1] value without listing the array:

```
admin> set read-access-hosts 1 10.2.3.4/24
```

Dependencies: You must set the Enforce-Address-Security parameter to Yes in the SNMP profile for the Read-Access-Hosts setting to have any effect.

Location: SNMP

See Also: Enforce-Address-Security, Read-Community, Read-Write-Community, Write-Access-Hosts

Write-Access-Hosts

Description: An array specifying up to eight IP addresses of SNMP managers with Write permission. The MAX TNT unit responds to SNMP Set, Get, and Get-Next commands from only the SNMP managers you specify.

Usage: Each element in the array can specify an IP address. With SNMP as the working profile, use the List command to display the array elements. You can then set a value in the Write-Access-Hosts parameter by specifying its numeric index and entering an address.

Example: To list the array elements and set the Write-Access-Hosts[1] value:

```
admin> list write-access-hosts
[in SNMP:write-access-hosts]
write-access-hosts[1] = 0.0.0.0
write-access-hosts[2] = 0.0.0.0
write-access-hosts[3] = 0.0.0.0
write-access-hosts[4] = 0.0.0.0
write-access-hosts[5] = 0.0.0.0
write-access-hosts[6] = 0.0.0.0
write-access-hosts[7] = 0.0.0.0
write-access-hosts[8] = 0.0.0.0
admin> set 1 10.2.3.4/24
```

To set the Write-Access-Hosts[1] value without first listing the array:

```
admin> set write-access-hosts 1 10.2.3.4/24
```

Dependencies: For the Write-Access-Hosts setting to restrict read-write access to the MAX TNT unit, you must set the Enforce-Address-Security parameter to Yes in the SNMP profile.

Location: SNMP

See Also: Enabled, Enforce-Address-Security, Read-Access-Hosts, Read-Community, Read-Write-Community

SNMP: MIB support for DS3-ATM version 2 and E3-ATM slot cards

TAOS release 9.0 includes support for SNMP management of the new DS3-ATM version 2 and E3-ATM slot cards. You can view or download the DS3-ATM version 2 and E3-ATM slot card guides at `http://www.lucent.com/ins/doclibrary/library.html`.

The following MIBs are supported in this release:

- Chassis MIB (Lucent proprietary)
- Interface MIB (RFC 2863)
- ATM MIB (RFC 1695, RFC 2515)
- DS3 MIB (RFC 1407, RFC 2496)
- mibe3atmnet.mib

SNMP: DS3 MIB support for the channelized T3 slot card

Currently, the DS1 MIB supports channelized T3 slot cards for individual T1 lines on the T3 card. This release introduces DS3 MIB support for channelized T3 slot cards.

New values for the DS3 MIB

The MAX TNT now supports the following new values in the DS3 MIB for the Set commands:

- `dsx3LineType`
- `dsx3CircuitIdentifier`
- `dsxLoopbackConfig`

The T3 card has a new interface in the ifTable with the following values:

- `ifDescr—Channelize T3 Slot` *`slot/item`*
- `ifType—ds3(30)`
- `ifspeed—44736000`
- `ifName—ds3` *`shelf-slot-item`*
- `ifHighSpeed—45`
- `ifLinkUpDownTrapEnable—enabled(1)`
- `ifConnectorPresent—true(1)`

Note: To get these interface entries into the ifTable, enter the `slot -r` command to restart the T3 card and then initialize the slot again.

A link up/down trap is generated for the T3 line whenever the DS3 interface goes up or down.

New parameter in the T3 profile

The following new TAOS command-line interface parameter has been added to the T3 profile in the TAOS 9.0 release.

Loopback

Description: Specifies which option to use for a loopback test on the interface.

Note: Normal data traffic is interrupted while the interface is looped back.

Usage: Specify one of the following settings:

- `No-Loopback` (default value)—No loopback.
- `Line-Loopback`—Loop the DS3 outwards (downstream).
- `Local-Loopback`—Loop the DS3 inwards (internally).

Example: **`set Loopback = No-Loopback`**

Location: T3 {shelf-*N* slot-*N* N}

New parameters in the T3-Stat profile

The following new parameters have been added to the T3-Stat profile in the TAOS 9.0 release to support the channelized T3 slot card.

Loss-Of-Signal

Description: Indicates a loss of signal on the line.

Usage: Possible values:

- `False`—No loss of signal.
- `True`—Loss of signal.

Example: **`loss-of-signal = False`**

Location: T3-Stat {shelf-*N* slot-*N* N}

Loss-of-Frame

Description: Indicates a loss-of-frame signal on the line (also known as a Red Alarm).

Usage: Possible values:

- `False`—No loss of frame.
- `True`—Loss of frame.

Example: **`loss-of-frame = False`**

Location: T3-Stat {shelf-*N* slot-*N* N}

Yellow-Receive

Description: Indicates that the MAX TNT unit is receiving a loss-of-frame signal from the remote end (also known as a Yellow Alarm).

Usage: Possible values:

- `False`—No loss of frame from the remote end.
- `True`—Loss of frame from the remote end.

Example: **`yellow-receive = False`**

Location: T3-Stat {shelf-*N* slot-*N* N}

Ais-Receive

Description: Indicates that the remote end is sending an alarm indication signal.

Usage: Possible values:

- `False`—No alarm indication signal from the remote end.
- `True`—Alarm indication signal from the remote end.

Example: `ais-receive = False`

Location: T3-Stat {shelf-*N* slot-*N* N}

See Also:

SNMP: Support for SMIv2 syntax

In this release, the Simple Network Management Protocol version 2 (SNMPv2) MIB syntax has been modified to support Structure of Management Information version 2 (SMIv2).

New files

The following new SNMPv2 MIB files have been added to support SMIv2 syntax:

- `rfc2578.mib` (SNMPv2-SMI)
- `rfc2579.mib` (SNMPv2-TC)

Changed MIBs

The following MIBs have been modified:

- `ianaiftype.mib`. IANAifTYpe has been changed to TEXTUAL-CONVENTION.
- `rfc1696.mib`. The following fields are no longer accessible:
  - `mdmIndex`
  - `mdmLineCapabilitiesIndex`
  - `mdmCCStoredDialStringIndex`
- `rfc1580.mib`: This file has been split into two files:
  - `rfc1850.mib` (OSPF-MIB)
  - `rfc1850_2.mib` (OSPF-TRAP-MIB)
- `rfc2863.mib`. The InterfaceIndex and InterfaceIndexOrZero fields have been changed to TEXTUAL-CONVENTION.
- `rfc2514.mib` and `rfc2571.mib`. These two files have been switched back to the original standard SMIv2 RFC MIBs.

SNMP: Support for the L2TP MIB

This release introduces the SNMP MIB for the Layer 2 Tunneling Protocol (L2TP).

Based on the Internet draft `draft-ietf-pppext-l2tp-mib-05`, the L2TP MIB is contained in the Ascend private MIB,

`iso.org.dod.internet.private.enterprises.ascend`, using the identifier `tunnelGroup.asndL2tp`.

TAOS 9.0 implementation of this MIB has the following limitations:

- Some variables are currently unavailable.
- The `TunnelIfIndex` currently has no related interface in the interface MIB.
- Some counters return a zero.

The following portions of the MIB are implemented in this release as read-only:

`l2tpConfig:`

- `-l2pAdminState`

`l2tpStats:`

- `l2tpProtocolVersion`
- `l2tpVendorName`
- `l2tpFirmwareRevision`

`l2tpDomainStatsTable:`

- `l2tpDomainStatsIdentifier`
- `l2tpDomainStatsTotalTunnels`
- `l2tpDomainStatsFailedTunnels`
- `l2tpDomainStatsFailedAuthentications`
- `l2tpDomainStatsActiveTunnels`
- `l2tpDomainStatsTotalSessions`
- `l2tpDomainStatsFailedSessions`
- `l2tpDomainStatsActiveSessions`

The remaining counters are currently returned as zero:

`l2tpTunnelStatsTable:`

- `l2tpTunnelStatsIfIndex`
- `l2tpTunnelStatsLocalTID`
- `l2tpTunnelStatsRemoteTID`
- `l2tpTunnelStatsState`
- `l2tpTunnelStatsInitiated`
- `l2tpTunnelStatsRemoteHostName`
- `l2tpTunnelStatsRemoteVendorName`
- `l2tpTunnelStatsRemoteFirmwareRevision`
- `l2tpTunnelStatsRemoteProtocolVersion`
- `l2tpTunnelStatsInitialRemoteRWS`
- `l2tpTunnelStatsBearerCapabilities`
- `l2tpTunnelStatsFramingCapabilities`
- `l2tpTunnelStatsTotalSessions`
- `l2tpTunnelStatsActiveSessions`

`l2tpSessionStatsTable:`

- `l2tpSessionStatsTunnelIfIndex`
- `l2tpSessionStatsLocalCID`

```
- l2tpSessionStatsRemoteCID
- l2tpSessionStatsUserName
- l2tpSessionStatsState
- l2tpSessionStatsCallType
- l2tpSessionStatsCallSerialNumber
- l2tpSessionStatsTxConnectSpeed
- l2tpSessionStatsRxConnectSpeed
- l2tpSessionStatsCallBearerType
- l2tpSessionStatsFramingType
- l2tpSessionStatsDNIS (*)
- l2tpSessionStatsCLID (*)
- l2tpSessionStatsSubAddress (*)
- l2tpSessionStatsPrivateGroupID (**)
- l2tpSessionStatsProxyLcp
- l2tpSessionStatsAuthMethod
- l2tpSessionStatsSequencingState
(*)  LNS only
(**) not available at this time in TAOS
```

SNMP: Support for the Remote Ping MIB

This release includes support for the Remote Ping MIB as specified by the Internet Engineering Task Force's (IETF's) Distributed Management Group.

Ping MIBs allow the creation of Ping tests that can be set up to periodically issue a series of operations and generate traps or event notifications to report test results.

Supported tables

In this release, the MAX TNT unit supports the following tables in the Remote Ping MIB:

- Ping Control Table (`pingCtlTable`)
- Ping Results Table (`pingResultsTable`)

Supported traps

The MAX TNT unit supports the following traps (event notifications) in the Remote Ping MIB:

- `pingProbeFailed`. Generated when a probe failure is detected.
- `pingTestFailed`. Generated when a Ping test is determined to have failed.
- `pingTestCompleted`. Generated at the completion of a Ping test.

Changes to the Remote Ping MIB

The following changes are made to the standard MIB:

- All the definitions have been changed for compliance with Simple Network Management Protocol version 1 (SNMPv1) Structure of Management Information (SMI).
- The syntax MAX-ACCESS has been changed to ACCESS for all the fields.
- All the MIB fields that had a STATUS value of Current have been assigned the Mandatory value instead.
- Fields with read-create access were changed to read-write.

Currently, you cannot modify the following variables, so they have been changed to read-only:

- `pingMaxConcurrentRequests`
- `pingCtlDataFill`
- `pingCtlMaxRows`
- `pingCtlStorageType`
- `pingCtlType`
- `pingCtlIfIndex`
- `pingCtlByPassRouteTable`

Unsupported features

Currently, the `pingProbeHistoryTable` in the Remote Ping MIB is not supported.

SNMP: Support for the ISDN Type of Number MIB

TAOS release 9.0 for the MAX TNT includes SNMP support for the ISDN Type of Number MIB.

The calling party number information element (IE), in the ISDN call setup message contains the field ISDN Type of Number. This field contains one of the following values, depending on the calling party number:

0 = Unknown
1 = International
2 = National
3 = Network Specific
4 = Subscriber
6 = Abbreviated
7 = Reserved

The following entry has been created in the following MIB to hold the value of the calling party number type:

`Lucent.eventGroup.eventTable.eventEntry`

SNMP: Support for SNMPv3 USM privacy

MAX TNT units now support privacy for SNMPv3 user-based security model (USM) authentication. SNMPv3 USM is described in RFC 2574, *User-based Security Model for SNMPv3*.

In this release, you can configure both authentication and privacy according to the SNMPv3 user-based security model (USM). Previously, data present in the unencrypted protocol data unit (PDU) could be copied and interpreted by unauthorized listeners on the wire. Privacy support remedies this situation.

Enabling privacy causes the MAX TNT unit to accept encrypted requests from the manager and send responses in encrypted format. The encryption uses a 64-bit Data Encryption Standard (DES) algorithm. The system uses the user's privacy password for generating the private key for the encryption.

SNMPv3 USM features

SNMPv3 security management provides MAX TNT units with the following management features, which use the SNMPv3 user-based security model (USM):

SNMP3 USM feature	Description
• Authentication	Provides data integrity and data origin authentication. The message authentication is coded with either the MD5 or the SHA hash function.
• Privacy	Protects messages from being copied and interpreted by unauthorized listeners on the network. Privacy is new in TAOS 9.0.
• Timeliness	Protects against message delay or replay.
• Discovery	Allows one SNMP engine to obtain sufficient information about the MAX TNT units' SNMP engine to establish communication between an SNMP manager station and the MAX TNT units.
• GetBulkRequest	Added from SNMPv2 to allow the SNMPv3 manager to minimize the number of protocol exchanges required to retrieve a large amount of management information. The GetBulkRequest protocol data unit (PDU) allows an SNMPv3 manager to request as large a response as possible.

Command-line interface changes

To support SNMPv3 USM privacy, this release adds a new value to the Security-Level parameter in the SNMP profile, and the four new parameters to the SNMPV3-USM-User profile.

Changes to the Security-Level parameter

The Security-Level parameter in the SNMP profile specifies the security level of the SNMP agent when SNMPv3 is in use.

When configuring SNMPv3 USM privacy support, set the Security-Level parameter in the SNMP profile to `auth-priv` (the new setting). When you specify this setting, all user transmissions with a security level of None or Auth-NoPriv are rejected with the error message `Unsupported Security Level`.

SNMPV3-USM-User profile

The SNMPv3-USM-USER profile provides the ability to create and edit users profiles. You must configure the following new parameters in this profile for SNMPv3 privacy support:

- Auth-Key
- Priv-Key
- Priv-Protocol

Following is an example of the relevant parameters in an SNMP-USM-User profile.

```
admin> new snmpv3 testv3
SNMPv3-USM-USER/testv3 read
admin> list
[in SNMPv3-USM-USER/testv3]
name* = testv3
active-enabled = yes
read-write-access = no(*)
auth-protocol = md5-auth(*)
priv-protocol = no-priv(*)
auth-key = (*)
priv-key = (*)
```

Note: (*) This symbol represents a factory default value setting.

Parameter	Specifies
Auth-Key	Specifies an authentication key for SNMPv3 USM users. In most cases, you do not set this string directly. Instead, use the snmpAuthPass command to generate the value. If you have permission to view passwords, the authentication key appears as a string with escape sequences for save and restore purposes. Otherwise, the authentication key appears as a row of asterisks. The default is null. • If you change the value of Auth-Key directly, keep in mind that the length of the escape sequence must be 10 (16d in hexadecimal) if Message Digest 5 (MD5) is in use and 14 (20d in hexadecimal) if the Secure Hash Algorithm (SHA) is in use. If you specify an invalid value, the unit uses the previous key, if any, to communicate with the SNMP manager. If no previous key exists, this USM user cannot communicate with the network until a valid key is set by means of the snmpAuthPass command. Suppose you use the snmpAuthPass command to generate the following 16-byte string: `27 0a dc 75 f8 98 e5 7c 4c 03 22 7d dd ac 0d ef`

Parameter	Specifies
	The system displays it as the following Auth-Key value: `'\x0a\xdcu\xf8\x98\xe5|L\x03"}\xdd\xac\x0d\xef` Consider the following: • You must generate the authentication key by means of the snmpAuthPass command before the SNMPV3-USM-User profile can be used for communication with the SNMP manager. • If you change the authentication protocol from MD5 to SHA (or vice versa), you must change the authentication key by means of the snmpAuthPass command. The previous protocol-and-key combination is used until you specify a new one. • If Auth-Protocol is No-Auth, Auth-Key does not apply.

Priv-Key	Specifies a privacy key for SNMPv3 USM users.
	In most cases, you do not set this string directly. Instead, use the snmpPrivPass command to generate the value. If you have permission to view passwords, the privacy key appears as a string with escape sequences for save and restore purposes. Otherwise, the privacy key appears as a row of asterisks. The default is null.
	If you change the value of Priv-Key directly, keep in mind that the length of the escape sequence must be 10 (16d in hexadecimal) if Message Digest 5 (MD5) is in use and 14 (20d in hexadecimal) if the Secure Hash Algorithm (SHA) is in use. If you specify an invalid value, the unit uses the previous key, if any, to communicate with the SNMP manager. If no previous key exists, this USM user cannot communicate with the network until a valid key is generated by means of the snmpPrivPass command.
	Suppose you use the snmpPrivPass command to generate the following 16-byte string:
	`27 0a dc 75 f8 98 e5 7c 4c 03 22 7d dd ac 0d ef`
	The system displays it as the following Priv-Key value:
	`'\x0a\xdcu\xf8\x98\xe5|L\x03"}\xdd\xac\x0d\xef`
	Consider the following:
	• You must generate the privacy key by means of the snmpPrivPass command before the SNMPV3-USM-User profile can be used for communication with the SNMP manager.
	• If you change the authentication protocol from MD5 to SHA (or vice versa), you must change the privacy key by means of the snmpPrivPass command. The previous protocol and key combination is used until you specify a new one.
	• If Priv-Protocol is No-Auth, Priv-Key does not apply.

Parameter	Specifies
Priv-Protocol	Enable/disable encryption of messages sent on behalf of the user to or from the SNMP engine, and if enabled, the type of privacy protocol to be used. Default setting is `no-priv`. Following are the valid values: • `no-Priv` (the default) specifies that no encryption is required and that privacy is disabled. • `des-priv` specifies that DES-based privacy is required. Incoming messages that are DES-encrypted are interpreted, and outgoing responses are encrypted using DES. Note that outgoing reports are not encrypted.

New commands for SNMPv3 USM

The following new commands have been added to support SNMPv3 USM:

- snmpAuthPass
- snmpPrivPass

A description of each new command follows.

snmpAuthPass

Description: Generates the authentication key of an SNMPv3 USM user.

Permission level: Update

Usage: **`snmpAuthPass`** ***`username password`***

Argument	Description
`username`	SNMPv3 USM user for whom an authentication key is generated.
`password`	Password for generating the authentication key.

The snmpAuthPass command can accept a username in escape sequence format.

Example: To generate the authentication key of the user `robin` with the password `abc123`:

```
admin> snmpAuthPass robin abc123
```

Dependencies: The password you specify is not stored in the system. It is used to generate an authentication key when the user is authenticated. The key is stored in the system.

See Also: snmpPrivPass

snmpPrivPass

Description: Generates the privacy key of an SNMPv3 USM user.

Permission level: Update

Usage: `snmpPrivPass` *`username password`*

Argument	Description
`username`	SNMPv3 USM user for whom a privacy key is generated.
`password`	Password for generating the privacy key.

The snmpPrivPass command can accept a username in escape sequence format.

Example: To generate the privacy key of the user `robin` with the password `abc123`:

```
admin> snmpPrivPass robin abc123
```

Dependencies: The password you specify is not stored in the system. It is used to generate a privacy key when the user is authenticated. The key is stored in the system.

See Also: snmpAuthPass

Example of SNMPv3 USM configuration

To configure the USM features for a user, you must specify a name for the profile and set the Active-Enabled parameter to `yes`. You must also specify a password if the Auth-Protocol parameter is set to anything but `no-auth`.

The following commands specify the use of MD5 authentication for messages sent on behalf of a user named `testv3` to or from the SNMP engine. The user is assigned read-write access to the unit's MIBs.

```
admin> new snmpv3-usm-user testv3
SNMPV3-USM-USER/testv3 read
admin> set active-enabled = yes
admin> set read-write-access = yes
admin> set priv-protocol = des-priv
admin> write
SNMPV3-USM-USER/testv3 written

admin> snmpAuthPass testv3 abc123
admin> snmpPrivPass testv3 abc123

admin> read snmpv3-usm-user testv3
SNMPV3-USM-USER/testv3 read
admin> list
[in SNMPV3-USM-USER/testv3]
name* = testv3
active-enabled = yes
read-write-access = yes
auth-protocol = md5-auth
priv-protocol = des-priv
auth-key = \xfd\xcd\xb2V\xa7\x81\xa7\x89n"\xd5\x02\x8b\xb2\xe7K
priv-key = \xfd\xcd\xb2V\xa7\x81\xa7\x89n"\xd5\x02\x8b\xb2\xe7K
```

Example of agent restriction to SNMPv3

The following commands cause the SNMP agent to use only SNMPv3 and to check a user's security level before allowing access:

```
admin> read snmp
SNMP read
```

```
admin> set snmp-message-type = v3-only
admin> set security-level = auth-nopriv
admin> write
SNMP written
```

Ascend SNMP-Framework and SNMP-User-Based MIBs groups

The SNMP-Framework and SNMP-User-Based MIBs are registered with the Internet Assigned Numbers Authority (IANA).

RFC 2571: SNMP-Framework MIB groups

The SNMP-Framework-MIB consists of the following SNMP engine groups:

SNMP group	Description
`snmpEngineID`	An SNMP engine's unique administrative identifier. • **Syntax:** SnmpEngineID • **Access:** Read-only • **Status:** Mandatory
`snmpEngineBoots`	The number of times that the SNMP engine has started or restarted itself since the snmpEngineID was last configured. • **Syntax:** Integer (1 to 2147483647) • **Access:** Read-only • **Status:** Mandatory
`snmpEngineTime`	The number of seconds since the value of the snmpEngineBoots object last changed. If incrementing this objects value exceeds the maximum, snmpEngineBoots is incremented as if a restart occurred and the value reverts to zero. • **Syntax:** Integer (0 to 2147483647) • **Access:** Read-only • **Status:** Mandatory
`snmpEngineMaxMessageSize`	The maximum length in octets of an SNMP message that this SNMP engine can send, receive, or process. The message length is determined by message size values supported by all of the transports available by the engine. • **Syntax:** Integer (484 to 2147483647) • **Access:** Read-only • **Status:** Mandatory

RFC 2574: SNMP-User-Based-SM MIB groups

The SNMP-User-Based-SM MIB consists of the following SNMP engine groups:

SNMP group	Description
`usmStatsUnsupportedSecLevels`	The total number of packets received by the SNMP engine that were dropped because the requested security level was either unknown or unavailable. • **Syntax:** Counter32 • **Access:** Read-only • **Status:** Mandatory
`usmStatsNotInTimeWindows`	This group provides information on the total number of packets received by the SNMP engine which were dropped because they appeared outside of the authoritative SNMP engine's window. • **Syntax:** Counter32 • **Access:** Read-only • **Status:** Mandatory
`usmStatsUnknownUserNames`	The total number of packets received by the SNMP engine that were dropped because they referenced a user that was unknown to the SNMP engine. • **Syntax:** Counter32 • **Access:** Read-only • **Status:** Mandatory
`usmStatsUnknownEngineIDs`	The total number of packets received by the SNMP engine that were dropped because they referenced an snmpEngineID that was unknown to the SNMP engine. • **Syntax:** Counter32 • **Access:** Read-only • **Status:** Mandatory
`usmStatsWrongDigests`	The total number of packets received by the SNMP engine that were dropped because they contained an unexpected digest value. • **Syntax:** Counter32 • **Access:** Read-only • **Status:** Mandatory
`usmStatsDecryptionErrors`	The total number of packets received by the SNMP engine that were dropped because they could not be decrypted. • **Syntax:** Counter32 • **Access:** Read-only • **Status:** Mandatory

SNMP: SNMP manager support for the USM MIB

The user-based security model (USM) MIB enables you to use a Simple Network Management Protocol (SNMP) manager to create, modify, and delete SNMPv3 USM users by means of Set and Get requests.

Overview of SNMPv3 USM MIB support

When you create, modify, and delete SNMPv3 USM users by means of SNMP Set and Get requests, you can perform the following tasks:

- Change user status from In-Service to Not In-Service, and vice-versa
- Modify authentication and privacy keys.
- Disable authentication and privacy.

However, you must use the local interface rather than SNMP to enable authentication and privacy. In addition, you cannot create or modify more than one user in the same SNMP request.

Creating, modifying, and deleting SNMPv3 USM users

This section provides a brief description of how to create, modify, and delete SNMPv3 USM users from an SNMP manager. Refer to RFC 2574 for complete details about how to create, modify, and delete entries in the USM User Table.

Consider the following:

- This implementation does not support the creation or modification of multiple users by means of a single request protocol data unit (PDU).
- The system processes only those PDUs that contain a combination of Set requests recommended by RFC 2574 to create, modify and delete users. For example, a PDU containing `SET-usmUserAuthProtocol` and `SET-usmUserAuthKeyChange` generates an error. These two requests must be sent in two separate PDUs in the appropriate order.

Creating an SNMPv3 USM user

This implementation allows any existing user to be used as a template. For this reason, Lucent recommends that you use the local interface to create a set of profiles for users with different security parameters. These profiles can, in turn, be used as templates. To create a new user, proceed as follows:

1. Clone a new user from a template that has appropriate security levels.
2. To specify privacy for the user, set the privacy key by using the `keyChange` value. Otherwise, set the protocol to `usmNoPrivProtocol`.
3. To specify authentication for the user, set the authentication key by using the `keyChange` value. Otherwise, set the protocol to `usmNoAuthProtocol`.
4. Activate the new user.

Modifying an SNMPv3 USM user

Use SNMP Get and Set requests to modify security levels, privacy and authentication protocols, privacy and authentication keys, and service status (In-Service to Not-In-Service, and vice versa).

If the password for the user is different from the password of the cloned user configuration, you must generate new keys from the configured password before the SNMP manager attempts to communicate with the MAX TNT unit. Failure to generate a proper authentication and/or privacy key results in an authentication error.

Deleting an SNMPv3 USM user

Delete an SNMPv3 USM user by setting `usmUserStatus` to `rowStatusDestroy`.

SNMP: Support for SNMPv3 notifications

The MAX TNT unit now authenticates and encrypts protocol data units (PDUs) as required by SNMPv3, and generates traps in SNMP version 2 (SNMPv2) Trap2 format. Depending on your configuration, a MAX TNT unit can send PDUs in SNMPv2 format or in pre-TAOS 9.0 format. You can specify the destinations for traps and the format of outgoing trap PDUs. In addition, two new MIBs—SNMP-TARGET-MIB and SNMP-NOTIFICATION-MIB—have been added.

SNMPv3 notifications support enables you to configure the MAX TNT unit to perform the following tasks:

- Send SNMPv1 traps (Trap PDUs) or SNMPv2 Traps (Trap2 PDUs).
- Send traps to a specified IP address and port.
- Send Trap2 PDUs with different levels of security.
- Send Trap2 PDUs with different user names.

The SNMPv3 notifications feature follows the specifications in RFC 2573.

Configuring SNMPv3 notifications support

To set up SNMPv3 notifications support, you configure the SNMPv3-Notification profile, and the SNMPv3-Target-Param profile.

You must also configure new and existing parameters in the Trap profile. To configure these parameters from the command-line interface (CLI), you must perform the following steps:

1 Create an SNMPv3-Notification profile, and set a tag to the profile.

2 Create an SNMPv3-Target-Param profile.

3 Set the message-processing model to the V1 or V3 option, and the security model to the V1 or V3-USM option. If the V3-USM option is selected, set the Security-Name parameter to a valid SNMPv3-USM-User profile name.

4 Create a Trap profile, and set the name, destination IP address, and destination port.

5 Add tag values to the Notify-Tag-List parameter. This tag list must contain tags that were set in the SNMPv3-Notification profile. Multiple tags can be configured in this tag list,

separated by spaces. Then, set Target-Params-Name to the name of the SNMPv3-Target-Param profile.

All the notification profiles in the system will find trap profiles with matching tags. The parameters in the trap profiles are used to send traps to the network.

Note: When you upgrade to software that supports the SNMPv3 notifications feature, the system automatically creates an SNMPv3-Notification profile and an SNMPv3-Target-Param profile; both are called `default`. Therefore, SNMPv1 traps configured in an earlier version of the software are still generated when you upgrade. You need not create new profiles. However, removing or modifying the `default` profiles might affect the transmission of SNMPv1 traps.

Configuring an SNMPv3-Notification profile

Following are the new SNMPv3-Notification parameters and their default settings:

```
admin> read snmpv3-notification
[in SNMPV3-NOTIFICATION/"" (new)]
name* = ""
active-enabled = no
tag = ""
type =
```

Parameter	Specifies
Name	Unique name for the profile, up to 16 characters.
Active-Enabled	Enable/disable generation of notifications. Yes specifies that the profile is used to generate notifications. No (the default) specifies that it is not used to generate notifications.
Tag	Value that links the SNMPv3-Notification profile with the Trap profile specifying the host address to which notification messages are sent. You can specify up to 255 characters. The default is null.
Type	*Not currently implemented.*

Configuring an SNMPv3-Target-Param profile

Following are the new SNMPv3-Target-Param profile parameters and their default settings:

```
[in SNMPV3-TARGET-PARAM/""]
name* = ""
active-enabled = no
msg-proc-model = v1
security-model = v1
security-name =
security-level = none
```

Parameter	Specifies
Name	Unique name for the profile, up to 16 characters. The default is null.
Active-Enabled	Enable/disable generation of notifications. Yes specifies that the profile is used to generate notifications. No (the default) specifies that it is not used to generate notifications.

Parameter	Specifies
Msg-Proc-Model	Message-processing model to use when generating SNMP messages. V1 (the default) specifies SNMP version 1. V3 specifies SNMP version 3. For SNMPv3 notifications support, specify V3.
Security-Model	Security model to use when generating SNMP messages. V1 (the default) specifies the SNMP version 1 security model. V3-USM specifies the SNMP version 3 User-Based Security Model (USM). For SNMPv3 notifications support, specify V3-USM. You can specify V1 only when you have also set Msg-Proc-Model to V1. You can specify V3-USM only when you set Msg-Proc-Model to V3. For the SNMPv3-Target-Param profile to have any effect when Security-Model is set to V3-USM, you must configure an SNMPv3-USM-User profile with the name specified for Security-Name parameter.
Security-Name	Security name that identifies the user on whose behalf SNMPv3 USM messages are generated. You can specify up to 22 characters. The default is null. Security-Name applies only if Security-Model is set to V3-USM.
Security-Level	Level of security to use when generating messages. None (the default) specifies no authentication and no privacy. Auth-NoPriv specifies authentication and no privacy. Auth-Priv specifies authentication and privacy. For Auth-Priv to apply, you must set the Priv-Protocol and Priv-Password parameters in the SNMPv3-USM-User profile.

Configuring a Trap profile

Following are the new parameters in the existing Trap profile and their default settings:

```
[in Trap/"" (new)]
active-enabled = yes
host-port = 162
inform-time-out =
inform-retry-count =
notify-tag-list = default
target-params-name = default
```

Parameter	Specifies
Active-Enabled	Whether traps are sent to the host specified by the profile. Yes (the default) specifies that traps are sent. No specifies that traps are not sent.
Host-Address	IP address to which traps are sent. The default is 0.0.0.0.
Host-Port	Port to which traps are sent. Specify a number from 1 to 65535. The default is 162.
Inform-Time-Out	*Not currently implemented.*
Inform-Retry-Count	*Not currently implemented.*
Notify-Tag-List	Space-separated list of the Tag value(s) in each SNMPv3-Notification profile.

Parameter	Specifies
Target-Params-Name	Value of the Name parameter in the SNMPv3-Target-Param profile, up to 22 characters.

Parameter reference

This section contains complete descriptions of each new parameter you use to configure SNMPv3 notifications.

Active-Enabled

Description: In an SNMPv3-Notifications or SNMPv3-Target-Param profile, specifies whether the profile is used to generate notifications. In a Trap profile, specifies whether traps are sent to the host specified by the profile.

Usage: Specify Yes or No.

- Yes specifies that the profile is used to generate notifications or that traps are sent.
- No (the default) specifies that the profile is not used to generate notifications or that traps are not sent.

Example: `set active-enabled = yes`

Location: SNMPv3-Notification *name*, SNMPv3-Target-Param *name*, Trap *name*

See Also: Host-Port, Msg-Proc-Model, Notify-Tag-List, Security-Level, Security-Model, Security-Name, Tag, Target-Params-Name

Host-Port

Description: Specifies the port to which traps are sent.

Usage: Specify a number from 1 to 65535. The default is 162.

Example: `set host-port = 20`

Location: Trap *name*

See Also: Active-Enabled, Msg-Proc-Model, Notify-Tag-List, Security-Level, Security-Model, Security-Name, Tag, Target-Params-Name

Msg-Proc-Model

Description: Specifies the message-processing model to use when generating SNMP messages.

Usage: Specify one of the following values:

- V1 (the default) specifies SNMP version 1.
- V3 specifies SNMP version 3. For SNMPv3 Notifications support, specify V3.

Example: `set msg-proc-model = v3`

Location: SNMPv3-Target-Param *name*

See Also: Active-Enabled, Host-Port, Notify-Tag-List, Security-Level, Security-Model, Security-Name, Tag, Target-Params-Name

Notify-Tag-List

Description: Specifies the tag list indicated by the Tag parameter value in each SNMPv3-Notification profile.

Usage: Specify the Tag value(s) you indicated in one or more SNMPv3-Notification profiles.

Example: `set notify-tag-list = default1`

Location: Trap *name*

See Also: Active-Enabled, Host-Port, Msg-Proc-Model, Security-Level, Security-Model, Security-Name, Tag, Target-Params-Name

Security-Level

Description: Specifies the level of security to use when generating messages.

Usage: Specify one of the following settings:

- None (the default) specifies no authentication and no privacy.
- Auth-NoPriv specifies authentication and no privacy.
- Auth-Priv specifies authentication and privacy.

Example: `set security-level = auth-priv`

Dependencies: For Auth-Priv to apply, you must set the Priv-Protocol and Priv-Password parameters in the SNMPv3-USM-User profile.

Location: SNMPv3-Target-Param *name*

See Also: Active-Enabled, Host-Port, Msg-Proc-Model, Notify-Tag-List, Security-Model, Security-Name, Tag, Target-Params-Name

Security-Model

Description: Specifies the security model to use when generating SNMP messages.

Usage: Specify one of the following values:

- V1 (the default) specifies the SNMP version 1 security model.
- V3-USM specifies the SNMP version 3 User-Based Security Model (USM). For SNMPv3 Notifications support, specify V3-USM.

Example: `set security-model = v3-usm`

Dependencies: Consider the following:

- You can specify V1 only when you have also set Msg-Proc-Model to V1.
- You can specify V3-USM only when you set Msg-Proc-Model to V3.
- When Security-Model is set to V3-USM, you must configure an SNMPv3-USM-User profile, with the name specified for the Security-Name parameter, in order for the SNMPv3-Target-Param profile to have any effect .

Location: SNMPv3-Target-Param *name*

See Also: Active-Enabled, Host-Port, Msg-Proc-Model, Notify-Tag-List, Security-Level, Security-Name, Tag, Target-Params-Name

Security-Name

Description: Specifies a security name that identifies the user on whose behalf SNMPv3 USM messages are generated.

Usage: Specify up to 22 characters. The default is null.

Example: `set security-name = newuser`

Dependencies: Security-Name applies only if Security-Model is set to V3-USM.

Location: SNMPv3-Target-Param *name*

See Also: Active-Enabled, Host-Port, Msg-Proc-Model, Notify-Tag-List, Security-Level, Security-Model, Tag, Target-Params-Name

Tag

Description: Specifies a value that links the SNMPv3-Notification profile with the Trap profile specifying the host address to which notification messages are sent.

Usage: Specify up to 255 characters. The default is null.

Example: `set tag = newtag`

Location: SNMPv3-Notification *name*

See Also: Active-Enabled, Host-Port, Msg-Proc-Model, Notify-Tag-List, Security-Level, Security-Model, Security-Name, Target-Params-Name

Target-Params-Name

Description: Specifies the value indicated by the Name setting in the SNMPv3-Target-Param profile.

Usage: Specify up to 22 characters.

Example: `set target-params-name = profile1`

Location: Trap *name*

See Also: Active-Enabled, Host-Port, Msg-Proc-Model, Notify-Tag-List, Security-Level, Security-Model, Security-Name, Tag

Changes to MIBs for SNMPv3 notifications support

The following sections describe changes to SNMP that support the notification feature.

New MIBs

Two new MIBs are defined in the files `rfc2573_1.mib` and `rfc2573_2.mib`:

```
snmpTargetMIB MODULE-IDENTITY
    ORGANIZATION "IETF SNMPv3 Working Group"
    DESCRIPTION
        "This MIB module defines MIB objects which provide
         mechanisms to remotely configure the parameters used
         by an SNMP entity for the generation of SNMP messages."
    REVISION    "9808040000Z"
    DESCRIPTION "Clarifications, published as
                 RFC2573."
    REVISION    "9707140000Z"
    DESCRIPTION "The initial revision, published as RFC2273."
    ::= { snmpModules 12 }


snmpNotificationMIB MODULE-IDENTITY
    ORGANIZATION "IETF SNMPv3 Working Group"
    DESCRIPTION
        "This MIB module defines MIB objects which provide
         mechanisms to remotely configure the parameters
         used by an SNMP entity for the generation of
         notifications."
    REVISION    "9808040000Z"
    DESCRIPTION "Clarifications, published as
                 RFC2573"
    REVISION    "9707140000Z"
    DESCRIPTION "The initial revision, published as RFC2273."
    ::= { snmpModules 13 }
```

The `snmpTargetMIB` contains `snmpTargetObjects`.

`snmpTargetObjects` contains

```
snmpTargetSpinLock
snmpTargetAddrTable
snmpTargetParamsTable
snmpUnavailableContexts
snmpUnknownContexts
```

`snmpTargetAddrTable` contains

```
snmpTargetAddrName
snmpTargetAddrTDomain
snmpTargetAddrTAddress
snmpTargetAddrTimeout
snmpTargetAddrRetryCount
snmpTargetAddrTagList
snmpTargetAddrParams
snmpTargetAddrStorageType
snmpTargetAddrRowStatus
```

`snmpTargetParamsTable` contains

```
snmpTargetParamsName
snmpTargetParamsMPModel
snmpTargetParamsSecurityModel
snmpTargetParamsSecurityName
snmpTargetParamsSecurityLevel
snmpTargetParamsStorageType
snmpTargetParamsRowStatus
```

`snmpNotificationMIB` contains `snmpNotifyObjects`.

`snmpNotifyObjects` contains `snmpNotifyTable`.

`snmpNotifyTable` contains

```
snmpNotifyName
snmpNotifyTag
snmpNotifyType
snmpNotifyStorageType
snmpNotifyRowStatus
```

Trap2 PDU formatTrap2 PDU format

If configured, Trap2 PDUs sent to the SNMP manager contain trap information as specified by RFC 1907, *Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)*. Trap object identifiers (OIDs) are sent instead of generic and specific integers. Trap object identifiers for generic traps are named as follows:

`snmpTraps.`*`specific trap ID`*

Object identifiers for enterprise traps are named as follows with the Ascend trap ID as defined in `ascendv3.trp`:

`ascend.ascendNotifications.`*`Ascend trap ID`*

Other specific traps are sent with object identifiers as defined in their respective MIBs.

Following are other common elements of the PDU and their values:

- `@ contextEngineID` is filled with `msgEngineID` because only one `snmpEngine` in the system is identified by `msgEngineID`.
- `@ contextName` is filled with an empty string to indicate the default context.

This PDU is encrypted as specified in the corresponding profile SNMPv3-Target-Param. But if the security name specified in the profile does not have a corresponding USM username in the system, outgoing PDUs are discarded and a log message with level `LOG_LEVEL_WARNING` generated.

SNMP: Saving and restoring encrypted configurations

You can now use an SNMP Set request to save and restore encrypted configuration data over the network by means of TFTP. Previously, you could not use SNMP to save and restore encrypted configuration data over the network.

Note: Sending a password in an unencrypted SNMP request defeats the primary purpose of encryption. Therefore, Lucent recommends that this feature be used only in combination with the SNMPv3 privacy feature.

Two new SNMP fields have been added to the `flashOperation` group of the Ascend Flash MIB. These fields can be used to specify encryption for saving and restoring configuration data.

flashOperationEncryptMethod

The `flashOperationEncryptMethod` field has been added to the Ascend Flash Mib flashGroup.

Description: Specifies the encryption method used to save the configuration data.

Usage: When `flashOperationEncryptMethod` is set to 1 (noEncryption1) the configuration data is saved in clear text. When the value is set to either 2 (md5Encryption2) or to 3 (desEncryption), the configuration data is saved with MD5 or DES encryption respectively. `flashOperationEncryptMethod` is not necessary to restore the configuration data. The encryption method is determined by the format of the configuration file.

flashOperationEncryptPassword

The `flashOperationEncryptPassword` field has been added to the Ascend Flash Mib flashGroup.

Description: Specifies the password key used to encrypt and decrypt configuration data.

Usage: The value of `flashOperationEncryptPassword` is used for the encryption and decryption of configuration data. If `flashOperationEncryptPassword` is not set, or is set to a null value, then no encryption is performed even if `flashOperationEncryptMethod` is set to MD5 or DES encryption. `flashOperationEncryptPassword` is ignored if `flashOperationEncryptMethod` is set to 1 (noEncryption).

SNMP: Support for the ifStackTable in the IF-MIB

This release includes Simple Network Management Protocol (SNMP) support for the `ifStackTable` in the IF-MIB as defined in RFC 2863, *The Interfaces Group MIB*.

Each entry in the `ifStackTable` defines a relationship between two entries in the `ifTable`. The entries indicate which layers run on top of other layers. The object `ifStackHigherLayer` contains the `ifIndex` value that corresponds to the higher layer, while `ifStackLowerLayer` contains the `ifIndex` value that corresponds to the lower layer.

Currently, the `ifStackTable` defines the relationship between DS1 interfaces and the DS3 interface on a channelized DS3-ATM trunk module. A DS3 (T3) interface can multiplex 28 DS1 (T1) inputs. Each DS1 interface is represented in the `ifTable`, along with the DS3 interfaces.

Definition of ifStackTable

Following is the `ifStackTable` definition from `rfc2863.mib`:

```
ifStackTable OBJECT-TYPE

SYNTAX      SEQUENCE OF IfStackEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION

   "The table containing information on the relationships between the
   multiple sub-layers of network interfaces. In particular, it
   contains information on which sub-layers run 'on top of' which
   other sub-layers. Each sub-layer corresponds to a conceptual row in
   the ifTable. For example, when the sub-layer with ifIndex value x
```

```
   runs over the sub-layer with ifIndex value y, then this table
   contains:

   ifStackStatus.x.y = active

   For each ifIndex value, I, which identifies an active interface,
   there are always at least two instantiated rows in this table
   associated with I. For one of these rows, I is the value of
   ifStackHigherLayer; for the other, I is the value of
   ifStackLowerLayer. (If I is not involved in multiplexing, then
   these are the only two rows associated with I.)

   For example, two rows exist even for an interface which has no
   others stacked on top or below it:

   ifStackStatus.0.x = active

   ifStackStatus.x.0 = active"
::= { ifMIBObjects 2 }
```

Stacking DS1 interfaces

Following is an example of how DS1 interfaces are stacked on DS3 interfaces. You need not represent DS3 interfaces, but if they are required, use the following stacking:

```
ifIndex  Description

1        DS3
2        DS1 #1
3        DS1 #2
4        DS1 #3
5        DS1 #4
6        DS1 #5
.        .
.        .
.        .
29       DS1 #28
```

Following are the `ifStackTable` entries:

```
HigherLayer   LowerLayer

2             1
3             1
4             1
5             1
6             1
.             .
.             .
.             .
29            1
```

SNMP: WAN line table now shows signaling type for T1 and E1 lines

In previous releases, the Simple Network Management Protocol (SNMP) WAN Line Table listed the line type for a WAN line. However, it did not list which of several possible signaling types the line used. Now, a new field enables you to determine the signaling type for a T1 or E1 line.

The wanLineSignaling field has been added to the wanLineTable object. If the wanLineType is wanTypeT1 or wanTypeE1, the value of the wanLineSignaling field is set to the signaling type for the T1 or E1 line. If the wanLineType is not wanTypeT1 or wanTypeE1, or if an error occurs when the unit attempts to retrieve the signaling information, the value `wan-signaling-other(1)` is returned for the wanLineSignaling field.

The wanLineSignaling field is a mandatory, read-only field. Following are the valid values for this field:

```
wanLineSignaling OBJECT-TYPE
    SYNTAX INTEGER {
        wan-signaling-other(1),
        wan-inband(2),
        wan-isdn-pri(3),
        wan-t1-pbx(4),
        wan-isdn-pri-nfas(5),
        wan-isdn-pri-tunnel(6),
        wan-e1-r2(7),
        wan-e1-korean(8),
        wan-e1-p7(9),
        wan-e1-chinese(10),
        wan-e1-metered(11),
        wan-e1-no-signaling(12),
        wan-e1-dpnss(13),
        wan-e1-argentina(14),
        wan-e1-brazil(15),
        wan-e1-philippine(16),
        wan-e1-indian(17),
        wan-e1-czech(18),
        wan-e1-malaysia(19),
        wan-e1-new-zealand(20),
        wan-e1-israel(21),
        wan-e1-thailand(22),
        wan-e1-kuwait(23),
        wan-e1-mexico(24),
        wan-ss7-gw(25),
        wan-ss7-robbed-bit(26),
        wan-r1-inband(27),
        wan-dtmf-r2(28),
        wan-fgd-in-fgd-out-inband(29),
        wan-fgd-in-fgc-out-inband(30),
        wan-fgc-in-fgc-out-inband(31),
        wan-fgc-in-fgd-out-inband(32)
```

```
}
ACCESS read-only
STATUS mandatory
DESCRIPTION
   "The signaling type for the T1/E1 line. The value
    wan-signaling-other(1) is returned if the wanLineType
    is not T1/E1 or if the T1/E1 profile is not found."
::= { wanLineEntry 19 }
```

Support for multiple requests in a call-logging packet

MAX TNT 9.0 provides a mechanism for delivering multiple requests in a single call-logging packet to a call-logging data receiver (such as NavisAccess) that supports the Lucent 16-bit vendor specific attributes.

This feature enables more efficient management of MAX TNT units with NavisAccess. Packing multiple call-logging requests into a single packet makes network-bandwidth and processor utilization more efficient.

The `call-log-multi-packet` parameter has been added to the Call-Logging profile. If this parameter is set to `yes`, then multiple call-logging requests are sent in a single packet.

Note: This parameter can be enabled only if `call-log-radius-compat` is set to `16-bit-vendor-specific`.

SNMP support for multiple request call-logging packets

To support the delivery of multiple requests in a single call-logging packet, the callLoggingMultiPacket field has been added to the `call_log.mib`.

```
callLoggingMultiPacket OBJECT-TYPE
SYNTAX         INTEGER {
                       enabled(1),
                       disabled(2)
                             }
ACCESSread-write
STATUSmandatory
DESCRIPTION     "When enabled,  multiple call logging requests
                   will be packed into a single packet.
                   This feature can be enabled only if
                   callLoggingRadCompatMode.0 =
                   rad16BitVendorSpecific(5)."
::= { callLoggingGroup 14 }
```

SNMP: Support for the NoResourceAvailable trap

The MAX TNT now supports the new SNMP trap (event notification) `NoResourceAvailable`, with a default value of `yes`, which has been added to the trap menu. Unless it is set to `no`, the trap is activated if a modem is not successfully allocated to a call.

To support this new trap, the following four new MIB variables are available in the telephone number indexed `dnisMgmtGlobalTable` table. In addition, the existing MIB variable `dnisGlobalPhoneNumber` in this table has been modified to accept telephone numbers as short as a single digit rather than the previous minimum value of 4 digits.

MIB variable	Specifies
`dnisGlobalCallsActive(8)`	The number of active calls using this Dialed Number Information Service (DNIS).
`dnisGlobalCallsResFailed(9)`	The number of calls that used this DNIS and failed due to limited resources such as modems. This counter is reset to zero the first time a call using this DNIS becomes active.
`dnisGlobalCallsContResFailed(10)`	The number of calls that used this DNIS and failed due to limited resources such as modems. This counter is never reset.
`dnisGlobalCallsFailed(11)`	The number of calls that used this DNIS and failed due to reasons other than limited resources such as modems.

The new trap also sends the following object identifier (OID):

```
[Ascend Enterprise MIB].26.1.4.1.9.[Integer A].[Integer(s) B]
```

`Integer A` is the number of digits in the telephone number of the call that failed to find a modem, and `Integer B` is the telephone number itself in period-separated format.

SNMP: Enhanced support for the sysSlotStateChange trap

In earlier releases, the sysSlotStateChange trap (event notification) included the slot index and the slot status, but not the slot serial number. This release enhances the sysSlotStateChangeTrap to include the slot serial number, in addition to the slot index and slot status.

In the a`scend.trp` file, the IMPORTS list corresponding to the slot state has been changed to read as follows:

```
slotIndex, slotStatus, slotSerialNumber
   FROM ASCEND-CHASSIS-MIB
```

The TRAP definition for sysSlotStateChange has been changed to read as follows:

```
sysSlotStateChange  TRAP-TYPE
ENTERPRISE      ascend
VARIABLES       { slotIndex, slotStatus, slotSerialNumber }
```

```
DESCRIPTION    "This trap is sent to all the managers in the slot group
                        when a slot card's SLOT-STATE profile is
                        created due to slot insertion, or the oper-
                        ational state transitions into a new state.
                        The new state is indicated by the included
                        value of slotStatus.

               Refer to the slotTable.slotIndex, slotTable.slotStatus
                        and slotTable.slotSerial Number descrip-
                        tions."

::= 22
```

sysSlotStateChange trap

When a MAX TNT slot stops working, starts working, or otherwise changes status, the sysSlotStateChange trap is sent out. In this trap, the last line corresponds to the serial number of slot 7:

```
Enterprise Specific Trap (22)

enterprises.ascend.slots.slotTable.slotEntry.slotIndex.7 = 7

enterprises.ascend.slots.slotTable.slotEntry.slotStatus.7 = operState-
Down(1)

enterprises.ascend.slots.slotTable.slotEntry.slotSerialNumber.7 =
932606973
```

SNMP: VoIP call jitter reporting

TAOS 9.0 implements an improved method of jitter calculation on the StrongARM (SARM) processor for reporting Real-Time Transport Protocol (RTP) packet transmissions. The packet jitter on a MAX TNT unit is reported to both the media gateway controller (such as Softswitch) for IP Device Control (IPDC) protocol packets and to NavisAccess administration systems.

The jitter calculation provides an estimate of the statistical variance of the RTP data packet interarrival time, measured in timestamp units and expressed as an unsigned integer. The interarrival jitter is defined to be the mean deviation (smoothed absolute value) of the difference in packet spacing at the receiver compared to the sender for a pair of packets. As the control digital signal processor (DSP) accepts packets from the i960 processor shared-memory interface, jitter is calculated using the formula defined in the RFC 1889, *RTP: A Transport Protocol for Real-Time Applications*, (Jan. 1996), IETF:

```
jitter += (1/16)*(d-(jitter))
```

where:

- `d` represents the difference between the current transit time and the previous transit time.
- `(jitter)` holds the estimated jitter value for the RTP packets

The results of this calculation returns the equivalent of the difference in the relative transit time for the two packets. The relative transit time is the difference between a packet's RTP time stamp and the receiver's clock at the time of arrival, measured in the same units. Since all time calculations on MAX TNT units are executed using a fixed-point system, the jitter calculation is implemented using the following modified version of the formula specified by RFC 1889:

```
jitter += d - (jitter + 8) >> 4);
```

In this case, d is the difference between the current transit time and the previous transit time, as defined in RFC 1889. This value is maintained for each slave DSP.

Transit time is calculated by calculating the time difference between two consecutive packets, and subtracting the difference in RTP time stamps. The values are reported to the i960 through the Query Call Stats message response, the 32-bit jitter response is added to the end of message as first word, upper 16 bits, second word, lower 16 bits. The value is the number of 125µ ticks.

SNMP: Support for VoIP call logging

TAOS 9.0 supports the collection of H.323 call information from MAX TNT units configured as MultiVoice gateways using SNMP administration systems. With this enhancement, a MAX TNT unit can now generate start records, stop records, and call progress records for both VoIP and fax calls.

H.323 call information from MAX TNT units performing VoIP call processing is provided to SNMP log clients. Each MAX TNT unit provides the following H.323 call information:

- Billing start records
- Billing stop records
- Call disconnect records
- Fax start records

Billing start records

A billing start record reports the point in the call where speech communications is established. Start records provide the following information:

Attribute	**Specifies**
Call ID	The H.323 protocol call ID. A unique nonzero number assigned by a MultiVoice gateway upon receipt of a call and is used to track all call processing events related to a particular call.
Conference ID	The H.225.0 protocol messaging ID. A unique nonzero number assigned by a MultiVoice gateway upon receipt of a call. Track H.225.0 call setup messages related to a particular call.
Dial time	The time a MultiVoice gateway waits to collect the dialed telephone number. This value is zero for calls originating from the LAN.
Setup time	The time from the moment a user finishes dialing the destination telephone number until the moment speech is established at the called destination.
Call origin	The IP address used to identify the calling origin. This can be the ingress MultiVoice gateway or an H.323-compliant terminal.
Remote IP	The IP address used to identify the called destination. The IP address can identify the egress MultiVoice gateway or an H.323-compliant terminal (PC).

Attribute	Specifies
Telephone number	The dialed number string entered by the user.
CLID number	The E.164 address associated with the calling origin.
Audio mode	The audio codec used to connect an H.323 call.

Billing stop records

A billing stop record reports the point in the call where speech communications terminates (a telephone goes on-hook). Stop records provide the following information:

Attribute	Specifies
Call ID	The H.323 protocol call ID. A unique nonzero number assigned by a MultiVoice gateway upon receipt of a call, and is used to track all call processing events related to a particular call.
Conference ID	The H.225.0 protocol messaging ID. A unique nonzero number assigned by a MultiVoice gateway upon receipt of a call. Track H.225.0 call setup messages related to a particular call.
Connect time	The time from the moment speech is established at the called destination until a caller hangs up (telephone goes on-hook).
Drop time	The moment when a call connection is dropped by the WAN or LAN connection, which ever signal is reported first.
Drop reason	The H.323 call drop reason. For normal call termination, the billing stop record reports `normalDrop`.

Call disconnect records

A call disconnect record is generated whenever a call is not terminated normally (such as, when a connection between endpoints is lost as a result of equipment failure or network failure). Disconnect records provide the following information; thought some information may not be present as depending upon the origin of the call failure:

Attribute	Specifies
Call ID	The H.323 protocol call ID. A unique nonzero number assigned by a MultiVoice gateway upon receipt of a call, and is used to track all call processing events related to a particular call.
Conference ID	The H.225.0 protocol messaging ID. A unique nonzero number assigned by a MultiVoice gateway upon receipt of a call. Track H.225.0 call setup messages related to a particular call.
Dial time	The time a MultiVoice gateway waits to collect the dialed telephone number. This value is zero for calls originating from the LAN.
Setup time	The time from the moment a user finishes dialing the destination telephone number until the moment speech is established at the called destination.
Call origin	The IP address used to identify the calling origin. This can be the ingress MultiVoice gateway or an H.323-compliant terminal.
Remote IP	The IP address used to identify the called destination. The IP address can identify the egress MultiVoice gateway or an H.323-compliant terminal (PC).

Attribute	Specifies
Telephone number	The dialed number string entered by the user.
CLID number	The E.164 address associated with the calling origin.
Audio mode	The audio codec used to connect an H.323 call.
Drop from	This identifies the location that disconnected the call, either WAN or LAN.
Drop reason	The H.323 call drop reason. For disconnect reports, this is an incomplete and interrupted call termination reason.

Fax start records

A fax start record is generated whenever a fax answer tone is detected during a VoIP. The fax record provides the following information:

Attribute	Specifies
Call ID	The H.323 protocol call ID. A unique nonzero number assigned by a MultiVoice gateway upon receipt of a call, and is used to track all call processing events related to a particular call.
Conference ID	The H.225.0 protocol messaging ID. A unique nonzero number assigned by a MultiVoice gateway upon receipt of a call. Track H.225.0 call setup messages related to a particular call.
Modulation type	The fax modulation type detected by the MultiVoice gateway (such as: V.21, V.27, V.29, V.17, etc.)
Speed	The transmission speed or modulation rate detected for this fax transmission by the MultiVoice gateway (such as: 2400, 4800, 7200, etc.)

Note: Fax records are generated only for T.38 fax transmissions. Transparent modem/fax calls will only generate billing start and stop records.

H.323 disconnect reasons

H.323 disconnect reasons have been added to `disconnect-reason-type.mibdef` for Ascend disconnect type. Reported disconnect reasons for standard and nonstandard call termination are recorded in Table 14.

Table 14. H.323 Call Drop Reason (reserve value from 500 to 700)

Call drop reason	Call drop code	Specifies
DIS_H323_DROP_REASON_NULL	500	Call drop reason not available
DIS_H323_DROP_REASON_NORMAL	501	Normal disconnect (caller hung up)
DIS_H323_DROP_REASON_DEST_BUSY	502	Called destination busy
DIS_H323_DROP_REASON_DEST_UNREACHABLE	503	Called destination unreachable

Call drop reason	Call drop code	Specifies
DIS_H323_DROP_REASON_REJECT	504	Call rejected by MAX TNT units
DIS_H323_DROP_REASON_WAN_FAILURE	505	WAN failure; egress MultiVoice gateway could not connect the call
DIS_H323_DROP_REASON_GATEWAY_RESOURCES	506	Egress MultiVoice gateway could not process the call
DIS_H323_DROP_REASON_NO_BANDWIDTH	507	Sufficient bandwidth not available on the LAN for this call
DIS_H323_DROP_REASON_GW_NOT_REGISTERED	508	Egress MultiVoice gateway is currently unregistered with the MultiVoice Access Manager
DIS_H323_DROP_REASON_INVALID_PIN	509	Caller entered an invalid Personal Identification Number
DIS_H323_DROP_REASON_INVALID_DNIS	510	Caller dialed invalid number for PIN
DIS_H323_DROP_REASON_NO_LAN_ANSWER	511	A LAN connection was not available
DIS_H323_DROP_REASON_STATE_MACHINE	512	Call state machine on MultiVoice gateway could not advance
DIS_H323_DROP_REASON_NO_LAN_DISCONNECT	513	The LAN dropped the connection
DIS_H323_DROP_REASON_FEGW_CAUSE_CODE	514	The egress MultiVoice gateway dropped the connection
DIS_H323_DROP_REASON_MAX_PIN_ATTEMPTS	515	The caller failed to authenticate on all attempts to enter their PIN
DIS_H323_DROP_REASON_CODER_DENIED	516	The MultiVoice gateway could not negotiate an audio codec selection with the far-end gateway

Support for encryption of configuration transferred through TFTP

In this release, the MAX TNT unit can use Data Encryption Standard (DES) encryption for saving or restoring a configuration over the network by means of Trivial File Transfer Protocol (TFTP).

You can save and restore an encrypted configuration by means of the Load and Save commands. Loading an encrypted configuration file onto a pre-9.0 MAX TNT running unit without encryption support generates no errors. All the encrypted configuration data is ignored.

Changes to the Save and Load commands

To support DES encryption of configuration data, the following option has been added to the Save command:

-e *encryption_type password*

Argument	Specifies
encryption_type	Encryption and decryption method . You can specify DES or MD5.
password	Password used to generate key for encryption and decryption.

When you use the Load command to restore the configuration, do not enter the *encryption_type* argument. Instead, specify the following on the Load command line:

-e *password*

The system restores the configuration by applying the same encryption it used to save it.

For example, to save a configuration in DES-encrypted format:

```
admin> save -e des john network 172.20.32.114 test.cfg
```

To restore the configuration:

```
admin> load -e john config network 172.20.32.114 test.cfg
```

Note: The `-e` option supports only a network target.

Error messages for DES support

The following error messages have been added to support DES encryption for configuration file transfer over TFTP:

Error	Meaning
`-e option:unknown` `encryption method` *`method`*	You specified an incorrect encryption method when you saved the configuration.
`1. File is corrupted, Encryption tag not found` `2. File is corrupted, Version tag not found`	The configuration file is corrupted.
`Wrong encryption password!!`	Configuration is encrypted but the password is incorrect.

Error	Meaning
`Configuration is encrypted`	Configuration is encrypted but no password was provided.
`Configuration is not encrypted!!`	Configuration is not encrypted but a password was provided.
`Encrypted protocol <ver> not supported!!`	Encryption version mismatch occurred.

Support for a maintenance state for slot cards

TAOS 9.0 now supports a slot card maintenance state that allows you to leave any MAX TNT slot card in a maintenance state. When a slot card is in maintenance state, it is completely inactive but can be monitored with the Show command. The slot card maintains visibility but does not generate any unnecessary errors. When a slot card is out of maintenance state it is active.

The slot card remains in or out of maintenance state until change it. Prior to this release, the system attempted to make slot cards operational. In this release, this feature allows the slot card state to remain the same through a system reset or reboot. As long as the card stays in the same slot, it starts in the same state (up, down, or maintenance) in which it was last configured.

Changes to the Slot and Show commands

To support the maintenance state, the command-line interface (CLI) Slot command has two new options. The Show command adds the maintenance operational state to its display of slot card status.

Slot Command

The **`slot`** command now includes the following new options.

Syntax element	Description
`-m`	Puts the slot in a maintenance state
`option` **`-all`**	Applies the specified slot command option to all the slot cards.

Any time the **`-u`** (bring up), **`-d`** (bring down), or **`-m`** options are invoked, a warning message is sent to the console as a reminder that the slot state change will be retained. for example

```
admin> slot -m 1
Slot 1/1, state change forced
warning: new state will remain until next explicit management action.
```

Similarly, if you enter a `slot -d` command, the affected slot card remains down even after a system reset.

Note: Any time a new slot card is installed in a slot, it starts up when the system reboots. Also, all cards return to an up state if the system nonvolatile RAM (NVRAM) is cleared.

Caution: If any errors occur during loading, such as missing load images or corrupted images, the loader brings down the slot card in question. You must to manually bring up the card by using the `slot -u` command, or by using a Set operation on the SNMP variable `slotAdminStatus`.

Show command

The `show` command is enhanced to display the maintenance state for slot cards. For example:

```
admin> show
Shelf 1 ( standalone ):
   Reqd                 Oper      Slot Type
  { shelf-1 slot-1 0 }MAINT MAINT 8t1-card
  { shelf-1 slot-3 0 }DOWN RESETmadd-card
  { shelf-1 slot-7 0 }UP  UP        10-unchan-t1-card
  { shelf-1 slot-9 0 }UP  UP        ether3-card
```

Syslog and SNMP changes

To support the new maintenance slot state feature, changes have been made to Syslog and the SNMP slotTable.

Syslog

Syslog records any administrative actions that change the slot state. A new fatal index (222) identifies when a slot card enters maintenance state, where that action took place (for example, the console or the remote IP address), and which User profile the command was entered from. For example:

```
SLOT CARD MAINTENANCE:   Index 222   Revision: 9.0a0e0 Slot 1/3
   Date: 04/07/2000. Time: 08:46:03
   Card put on maintenance by console, user profile super.
SLOT CARD MAINTENANCE:   Index 222   Revision: 9.0a0e0 Slot 1/3
   Date: 04/07/2000. Time: 08:46:03
   Card put on maintenance by 192.168.1.2, user profile super.
```

SNMP slotTable

The SNMP slotTable describes the values of the installed slots. Changes have been made to the slotStatus field, slotAdminStatus attribute, and sysSlotStateChange traps.

slotStatus

The slotStatus field displays the current status of the MAX TNT slot cards. This release adds support for the `operStateMaintenance` value for this field, which reflects a new possible operational state.

slotAdminStatus

This attribute changes the state of the slot card or deletes the slot card. When read, it displays the last administrative state that was set. The new `maintenance` value permits the state of a slot card to be changed to a maintenance state.

sysSlotStateChange trap

The sysSlotStateChange trap is generated whenever the operational state changes. This notification is sent to all of the managers in the slot group when a slot card's SLOT-STATE profile is created, or the operational state is changed. The new state is indicated by the value of slotStatus.

See "SNMP: Enhanced support for the sysSlotStateChange trap" on page 61 for more information about this trap.

NavisAccess: Enhanced network management

MAX TNT units now support optional levels of network management.

Overview

The types of network management that can be performed by NavisAccess depend on which functionality options you purchased. If necessary, you can purchase one or more additional network management options after installing NavisAccess. Previously, only the network management (NM) base feature was available.

The following two new network management options can now be enabled:

- Network management (NM) with Voice over IP (VoIP) enabled (NMV)
- High-density network management (NM+) with VoIP enabled (NM+V)

Table 15 illustrates the complete set of network management (NM) features, their names in the TAOS command-line interface (CLI) Base profile, and the associated line density each supports.

Table 15. Network management features for NavisAccess software

Network management feature	Name in Base profile	Line density supported	VoIP enabled	Comment
NM	`network-management-enabled`	Supports up to 4 DS3 lines.	No	Default feature
NMV	`network-mgmt-voip-enabled`	Supports up to 4 DS3 lines.	Yes	Same as NM, plus VoIP.
NM+V	`network-mgmt-plus-voip-enabled`	Supports up to 12 DS3 lines.	Yes	Same as NM+ plus VoIP.

Changes to the Base profile

You can read and list the Base profile to verify which NavisAccess network management features are enabled on your system. For example:

```
admin> read base
BASE read (read-only)
admin> list
[in BASE]
shelf-number = 1
software-version = 9
software-revision = 0
.
.
.
network-management-enabled = no
frvc-enabled = enabled
network-mgmt-voip-enabled = no
```

For additional information about optional Network Management functionality, see your Lucent reseller.

Rlogin and raw TCP support added to terminal-server menus

In previous releases, the terminal-server menu provided options only for Telnet and PPP sessions. Now, the menu can provide the user with option of beginning an Rlogin or raw TCP (TCP clear) session as well. In addition, you can now specify the PPP option anywhere in the menu. For Telnet, raw TCP, and Rlogin, the host authenticates the session. For PPP, the MAX TNT unit authenticates the session.

Changes to the command-line interface (CLI)

The Menu-Mode-Options subprofile of the Terminal-Server profile contains the following new parameters:

- Service-1, Server-2, Service-3, Service-4 for specifying the type of service associated with an option
- Port-1, Port-2, Port-3, Port-4 for specifying the port to use for Telnet sessions.
- User-1, User-2, User-3, User-4 for specifying a username for Rlogin sessions.

In addition, the Hosts-Info *N* subprofile of the Ext-Tsrv profile contains new User and Service parameters.

Changes to the Menu-Mode-Options subprofile

Following is a listing of the Menu-Mode-Options subprofile with the new parameters:

```
[in TERMINAL-SERVER:menu-mode-options]
start-with-menus = no
toggle-screen = no
remote-configuration = no
text-1 = ""
host-1 = 0.0.0.0
service-1 = telnet
```

```
port-1 = 0
user-1 = ""
text-2 = ""
host-2 = 0.0.0.0
service-2 = telnet
port-2 = 0
user-2 = ""
text-3 = ""
host-3 = 0.0.0.0
service-3 = telnet
port-3 = 0
user-3 = ""
text-4 = ""
host-4 = 0.0.0.0
service-4 = telnet
port-4 = 0
user-4 = ""
```

Following are descriptions of each new parameter.

Port-1

Description: Specifies the port to use for contacting the Telnet host specified by Host-1.

Usage: Specify a number from 0 to 65535. The default is 0 (zero).

Example: **`set port-1 = 50`**

Dependencies: Port-1 applies only when Service-1 is set to Telnet.

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-1, Service-1

Port-2

Description: Specifies the port to use for contacting the Telnet host specified by Host-2.

Usage: Specify a number from 0 to 65535. The default is 0 (zero).

Example: **`set port-2 = 50`**

Dependencies: Port-2 applies only when Service-2 is set to Telnet.

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-2, Service-2

Port-3

Description: Specifies the port to use for contacting the Telnet host specified by Host-3.

Usage: Specify a number from 0 to 65535. The default is 0 (zero).

Example: **`set port-3 = 50`**

Dependencies: Port-3 applies only when Service-3 is set to Telnet.

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-3, Service-3

Port-4

Description: Specifies the port to use for contacting the Telnet host specified by Host-4.

Usage: Specify a number from 0 to 65535. The default is 0 (zero).

Example: `set port-4 = 50`

Dependencies: Port-4 applies only when Service-4 is set to Telnet.

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-4, Service-4

Service-1

Description: Specifies the type of service to use for the host specified by Host-1.

Usage: Specify one of the following values:

- Telnet (the default) specifies Telnet service.
- RawTCP specifies raw TCP service.
- Rlogin specifies Rlogin service.
- PPP specifies PPP service.

Example: `set service-1 = rlogin`

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-1

Service-2

Description: Specifies the type of service to use for the host specified by Host-2.

Usage: Specify one of the following values:

- Telnet (the default) specifies Telnet service.
- RawTCP specifies raw TCP service.
- Rlogin specifies Rlogin service.
- PPP specifies PPP service.

Example: `set service-2 = rlogin`

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-2

Service-3

Description: Specifies the type of service to use for the host specified by Host-3.

Usage: Specify one of the following values:

- Telnet (the default) specifies Telnet service.
- RawTCP specifies raw TCP service.
- Rlogin specifies Rlogin service.
- PPP specifies PPP service.

Example: `set service-3 = rlogin`

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-3

Service-4

Description: Specifies the type of service to use for the host specified by Host-4.

Usage: Specify one of the following values:

- Telnet (the default) specifies Telnet service.
- RawTCP specifies raw TCP service.
- Rlogin specifies Rlogin service.
- PPP specifies PPP service.

Example: `set service-4 = rlogin`

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-4

User-1

Description: Specifies the username for Rlogin sessions with Host-1.

Usage: Specify a text string of up to 31 characters. The default is null.

Example: `set user-1 = robin`

Dependencies: User-1 applies only when Service-1 is set to Rlogin.

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-1, Service-1

User-2

Description: Specifies the username for Rlogin sessions with Host-2.

Usage: Specify a text string of up to 31 characters. The default is null.

Example: `set user-2 = robin`

Dependencies: User-2 applies only when Service-2 is set to Rlogin.

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-2, Service-2

User-3

Description: Specifies the username for Rlogin sessions with Host-3.

Usage: Specify a text string of up to 31 characters. The default is null.

Example: `set user-3 = robin`

Dependencies: User-3 applies only when Service-3 is set to Rlogin.

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-3, Service-3

User-4

Description: Specifies the username for Rlogin sessions with Host-4.

Usage: Specify a text string of up to 31 characters. The default is null.

Example: `set user-4 = robin`

Dependencies: User-4 applies only when Service-4 is set to Rlogin.

Location: Terminal-Server > Menu-Mode-Options

See Also: Host-4, Service-4

Changes to the Host-InfoN subprofile

Following are descriptions of the new User and Service parameters.

Service

Description: Indicates the type of service to use for the host.

Usage: The Service value is read only. It can be one of the following:

- Telnet indicates Telnet service.
- RawTCP indicates raw TCP service.
- Rlogin indicates Rlogin service.
- PPP indicates PPP service.

Example: `service = rlogin`

Location: Ext-Tsrv > Hosts-Info *N*

See Also: User

User

Description: Indicates the username for the Rlogin session.

Usage: The User value is read only.

Example: `user = robin`

Location: Ext-Tsrv > Hosts-Info *N*

See Also: Service

RADIUS changes for Rlogin and raw TCP support

The format of the string for the Ascend-Host-Info (252) attribute now enables you to specify the service type and username. Following is an updated description of this attribute.

Description: Specifies a list of hosts to which a user can establish a Telnet, Rlogin, or PPP session.

Usage: You can specify up to 10 Ascend-Host-Info entries in a user profile. Enter your setting in the following format:

`Ascend-Host-Info="[`*`service`*`][`*`username`*`] `*`IP_address`*`[:`*`port`*`] `*`text`*`"`

Argument	Description
service	Telnet, Rlogin, rawTCP, or PPP. Telnet is the default.
username	Username for an Rlogin session.
IP_address	IP address of each host.
:*port*	Port for contacting the Telnet host.
text	Description of each host, up to 31 characters.

The MAX TNT unit assigns each entry a number. When the user selects the number, the terminal server initiates a session with the host at the specified IP address.

Example: To set up a host list for a MAX TNT unit named `Cal`, you would configure a pseudo-user profile as follows:

```
banner-Cal Password = "ascend"
   Service-Type = Outbound
   Reply-Message    = "sp-max terminal server"
   Reply-Message    = "! You are welcome !"
   Reply-Message    = "      :-)          "
   Ascend-Host-Info = "telnet 200.167.61.39 telnet to apache"
   Ascend-Host-Info = "rawtcp 205.168.62.38:21 raw tcp service
   Ascend-Host-Info = "200.167.61.39:23 telnet to apache"
   Ascend-Host-Info = "rlogin ps 200.168.64.31 rlogin to apache"
   Ascend-Host-Info = "ppp PPP service"
```

See Also: Reply-Message (18) in the *TAOS RADIUS Guide and Reference*

New Diag command

In this release, the new Diag command enables centralized control of all debug output in the system.

In previous releases, many debug commands existed, and each enabled you to turn debug output on or off for a particular system component. In this release, the functions of many of these commands have been consolidated in a single new Diag command.

The sections that follow describe the basic uses of the Diag command.

Note: The Diag command options available depend on whether you type the command at the console or from a slot card. For those options specific to a particular slot card, you must open a session with the slot card before executing the command.

Generating a list of all system components for which to generate debug output

To generate a list of all the system components for which you can generate debug output, enter the following command:

```
admin> diag ?
```

Following is a partial list of the components that the system displays:

```
arp                          ( Address Resolution Protocol )
networki                     ( Call Control )
vrouter <0xffff>             ( Virtual Router )
vroutercb                    ( Control Bus )
xdb <0xff>                   ( Radius )
zip                          ( AppleTalk )
```

Enabling debug output for all system components

To enable debug output for all system components, enter the following:

```
admin> diag ALL
```

Listing all system components with debug output enabled

To list all system components with debug output enabled, enter the following:

```
admin> diag -l
```

Enabling or disabling debug output

To enable or disable debug output for a particular system component, enter the following command:

```
admin> diag component
```

The command works as a toggle. For example, to enable debug output for ARP, enter the following:

```
admin> diag arp
arp debug is ON
```

To disable ARP output, enter the following:

```
admin> diag arp
arp debug is OFF
```

Enabling debug output for components with output disabled

To enable debug output for any components for which output is currently disabled, enter the following:

```
admin> diag ON
```

For example, suppose that the `networki` and `zip` components are currently disabled. To enable them, enter the following:

```
admin> diag on
networki debug is ON
zip debug is ON
```

Disabling debug output for components with output enabled

To disable debug output for any components for which output is currently enabled, enter the following:

```
admin> diag off
```

For example, suppose that the `networki` and `zip` components are currently enabled. To disable them, enter the following:

```
admin> diag off
networki debug is OFF
zip debug is OFF
```

New options for the NSLookup command

The `-s` ***`DNS_server`*** and `-v` options have been added to the NSLookup command. Following is a full description of the NSLookup command with these new options.

NSlookup

Description: Resolves the IP address of a specified hostname or Virtual Router (VRouter) by performing a Domain Name System (DNS) lookup.

Permission level: Diagnostic

Usage: **`nslookup [-r`** ***`Vroutername`***`][-s`** ***`DNS_server`***`] [-v]`** ***`hostname`***

Syntax element	Description
`-r` ***`Vroutername`***	VRouter for which you want to obtain an IP address.
`-s` ***`DNS_server`***	Specifies the IP address of the DNS server that the MAX TNT unit uses to resolve the hostname or VRouter name. If you do not specify the `-s` option, the system uses the DNS server that you configured locally.
`-v`	Specifies that the unit prints the details of the packet received from the DNS server.
`hostname`	Hostname for which you want to obtain an IP address.

Example: To look up the IP address of `host-231` by means of the DNS server at 10.65.12.10:

```
admin> nslookup -s 10.65.12.10 host-231
Resolving host host-231.
IP address for host host-231 is 10.65.12.231.
```

Dependencies: Unless you use the `-s` option, your unit must be configured with the address of at least one DNS server.

See Also: ARPtable, Netstat

Displaying call session and authentication statistics

Call connections and authentication statistics can now be displayed on MAX TNT units. You enable the new `callstats-list` value by setting the `left-stat` parameter in the User profile. Prior to this release, the Left-Stat profile only provided the Session-List and Connection-List settings.

Parameter	Indicates
`left-stat`	Current system information appearing by default in the left side of the status window: `session-list`—Displays the current system administration sessions. `connection-list`—Displays current system WAN connections. `callstats-list`—Displays current system call statistics. These statistics include timed interval information on the number of calls connected and authenticated.

Displaying call connection and authentication statistics

The following examples show how to configure and display call connection and authentication statistics on the MAX TNT.

The following example shows how to configure call statistics by setting the `left-status` parameter to the `callstats-list` option in the User profile:

```
USER/admin read
admin> list
[in USER/admin]
name* = admin
admin> set left-status = callstats-list
admin> write
```

The following example shows how to display call statistics by entering `view left callstats` from the `admin>` prompt.

```
admin> view left callstats

------------------------------|
1250 Connections, 1250 Sessions|<Left Screen displays call statistics
Call Statistics for the last   |
------------------------------|
   10s  20s  30s  40s  50s  60s|<Calls display in 10 second intervals
------------------------------|
C   44   54   48   48   44   36|<Row 1 displays # of call sessions
A   61   44   43   37   54   53|<Row 2 shows # of calls authenticated
------------------------------|
    10m   20m   30m   40m   50m |<Row 2 shows # of calls authenticated

------------------------------|
C  2997  3003  3001  3013  3004|<Row 3, "1-10m" updated every minute
A  3006  3008  3009  2998  3011|<Row 4, "1-10m" updated every minute
```

```
------------------------------|
Total            1min      1hr |
------------------------------|
Connected         274    18009  |<Row 5 displays total calls connected
Authenticated     292    18031 |<Row 6 shows total calls authenticated
------------------------------
```

Row 5 of the CallStats-List shows a total of 274 calls connected in a 60-second time frame, and a total of 18,009 calls connected in a 60-minute time frame. Row 6 of the CallStats-List shows a total of 292 calls that have been authenticated in a 60-second time frame, and a total of 18,031 calls authenticated within a 60-minute time frame.

Support for the Load Tar command using multiple filenames

Some implementations of `tfpd`—servers that support the Trivial File Transfer Protocol (TFTP)—impose a file-size limitation of 16 MB. To ensure that sites using `tftp` to load TAOS software are not affected by this limitation, the syntax of the `load tar` command has been changed to allow multiple filenames. Following is the new syntax:

`load tar network` ***`host file1`*****`.tar`** `[`***`file2`*****`.tar`**`] [...] [`***`flash-card-id`***`]`

(Items enclosed in brackets in the syntax statement are optional.)

In addition, TAOS slot-card images are now provided in two tar files. The first tar file contains the most commonly used slot-card images, listed in Table 16.

Table 16. Contents of the first tar file

Filename	Contents	
	Description	**Slot-card images**
`tntrel.tar`	Shelf controller	`tntsr`
	Ethernet	`tntenet2 tntenet3 tntenet3nd`
	HDLC	`tnthdlc2 tnthdlc2ec`
	T1-specific images	`tnt8t1 tntt3 tntut1 tntpctfit`
	MAX TNT modem images	`tntcsmx tntcsm3v tnt-madd tntmdm56k`
`tntrele.tar`	Shelf controller	`tntsre`
	Ethernet	`tntenet2 tntenet3 tntenet3nd`
	HDLC	`tnthdlc2 tnthdlc2ec`
	E1-specific images	`tnt8e1 tntue1 tnt-pct-fie`
	MAX TNT modem images	`tntcsmx tntcsm3v tnt-madd tntmdm56k`

Table 16. Contents of the first tar file (continued)

Filename	Contents	
	Description	**Slot-card images**
apxrel.tar	Shelf controller	apxsr
	Ethernet	tntenet2 tntenet3 tntenet3nd
	HDLC	tnthdlc2 tnthdlc2ec
	T1-specific images	tnt8t1 tntt3 tntut1 tntpctfit
	APX 8000 modem images	tntcsmx tntcsm3v tnt-madd
apxrele.tar	Shelf controller	apxsre
	Ethernet	tntenet2 tntenet3 tntenet3nd
	HDLC	tnthdlc2 tnthdlc2ec
	E1-specific images	tnt8e1 tntue1 tnt-pct-fie
	APX 8000 modem images	tntcsmx tntcsm3v tnt-madd

The second tar file contains images for slot cards that are less commonly used, listed in Table 17.

Table 17. Contents of the second tar file

Filename	Contents	
	Description	**Slot-card images**
tntrel2.tar	T1-specific images	tntstm0 tntuds3 tntds3atm tntds3atm2 tntoc3atm
	MAX TNT modem	tntamdm
	Serial WAN	tntswan
tntrele2.tar	E1-specific images	tnte3atm
	MAX TNT modem	tntamdm
	Serial WAN	tntswan
apxrel2.tar	T1-specific images	tntstm0 tntuds3 tntds3atm tntds3atm2 tntoc3atm
	Serial WAN	tntswan
apxrele2.tar	E1-specific images	tnte3atm
	Serial WAN	tntswan

If the unit does not contain any of the slot cards supported in the second tar file, load only the first tar file. For example:

```
admin> load tar network 10.10.10.10 tntrel.tar
```

If the unit contains slot cards supported in the second tar image, both files *must* be loaded on the same command line. For example:

```
admin> load tar network 10.10.10.10 tntrel.tar tntrel2.tar
```

The system loads only the images required for slot cards installed in the system.

Warning: Do not load the second tar file alone. Loading the second tar file without the first tar file causes the system to delete necessary images from flash. If this occurs, enter the `load` command again, specifying both tar files on the command line.

Extensions features in TAOS 9.0

Global digital access extension features

WORM-ARQ for personal digital cellular phones

Nippon Telephone and Telegraph (NTT) DoCoMo has developed a technology called Window-control Operation based on Reception Memory-Automatic Retransmission Request (WORM-ARQ), which maintains transmission quality for personal digital cellular (PDC) wireless phones in Japan. With a MultiDSP card installed and a WORM-ARQ license enabled in TAOS, the MAX TNT unit supports calls of this type. Use the following commands to confirm that the WORM-ARQ license is enabled in your system:

```
[in BASE:wormarq-enabled]
wormarq-enabled = enabled
```

When the WORM-ARQ license is enabled, the system creates a new Call-Route profile for each installed MultiDSP card. The new Call-Route profile sets the Call-Route-Type parameter to `wormarq-call-type`, as shown in the following sample profile:

```
admin> get call-route { { { 1 12 0 } 0 } 2 }
[in CALL-ROUTE/{ { { shelf-1 slot-12  } 0 2 }]
index* = { { { shelf-1 slot-12 0 } 0 } 2 }
trunk-group = 0
telephone-number = ""
preferred-source = { { any-shelf any-slot 0 } 0}
call-route-type = wormarq-call-type
```

This Call-Route-Type setting enables the system to route WORM-ARQ calls to the MultiDSP cards in the system.

Support for PIAFS 2.1 on the 96-port MultiDSP card

TAOS 9.0 extends support for the PHS Internet Access Forum Standard (PIAFS) 2.1 to the 96-port MultiDSP slot card. In previous software releases, the MAX TNT unit supported PIAFS 2.1 only for the 48-port MultiDSP slot card.

With a 96-port MultiDSP card installed, the MAX TNT unit supports the PIAFS protocol required for the Personal Handyphone System (PHS). PHS service is currently available only with Japan PRI signaling. With PIAFS 2.1, the MAX TNT supports a data rate that switches between 32 Kbps and 64 Kbps during a call, depending on what the wireless bandwidth

permits. PIAFS version 2.1 has an enhanced link-level protocol that supports dynamic switching of data rates between 32 Kbps and 64 Kbps.

When the PHS-Support and PHS-2-1 licenses have been enabled, the system creates a new Call-Route profile for each installed MultiDSP card. The new Call-Route profile sets the Call-Route-Type parameter to `phs-call-type`, as shown in the following sample profile:

```
admin> get call-route { { { 1 12 0 } 0 } 2 }
[in CALL-ROUTE/{ { { shelf-1 slot-12  } 0 2 }}]
index* = { { { shelf-1 slot-12 0 } 0 } 2 }
trunk-group = 0
telephone-number = ""
preferred-source = { { any-shelf any-slot 0 } 0}
call-route-type = phs-call-type
```

This Call-Route-Type setting enables the system to route PHS calls to the card.

To display the PHS call routes on the 96-port MultiDSP slot card, use the `callroute` command, with `-ah` option to display routes by type or with the `-d` option to display routes by device, as in the following examples:

```
admin> callroute -ah
device      #            source              type   tg   sa phone
1:07:01/0   4 0:00:00/0 v110-call-type     0      0
1:07:02/0   4 0:00:00/0 v110-call-type     0      0
1:07:03/0   4 0:00:00/0 v110-call-type     0      0
1:07:04/0   4 0:00:00/0 v110-call-type     0      0
1:07:96/0   4 0:00:00/0 v110-call-type     0      0
1:07:01/0   3 0:00:00/0 voip-call-type     0      0
1:07:02/0   3 0:00:00/0 voip-call-type     0      0
1:07:01/0   2 0:00:00/0 phs-call-type      0      0
1:07:02/0   2 0:00:00/0 phs-call-type      0      0
1:07:03/0   2 0:00:00/0 phs-call-type      0      0
1:07:04/0   2 0:00:00/0 phs-call-type      0      0
1:07:05/0   2 0:00:00/0 phs-call-type      0      0
1:07:06/0   2 0:00:00/0 phs-call-type      0      0
1:07:95/0   2 0:00:00/0 phs-call-type      0      0
1:07:95/0   1 0:00:00/0 digital-call-type  0      0
1:07:96/0   1 0:00:00/0 digital-call-type  0      0


admin> callroute -d
device    #             source              type  tg     sa phone
1:07:01/0 0 0:00:00/0   voice-call-type     0     0
1:07:01/0 1 0:00:00/0   digital-call-type   0     0
1:07:01/0 2 0:00:00/0   phs-call-type       0     0
1:07:01/0 3 0:00:00/0   voip-call-type      0     0
1:07:01/0 4 0:00:00/0   v110-call-type      0     0
```

```
1:07:01/0 5 0:00:00/0   any-call-type      0    0
1:07:02/0 0 0:00:00/0   voice-call-type    0    0
1:07:02/0 1 0:00:00/0   digital-call-type  0    0
1:07:02/0 2 0:00:00/0   phs-call-type      0    0
1:07:02/0 3 0:00:00/0   voip-call-type     0    0
1:07:02/0 4 0:00:00/0   v110-call-type     0    0
1:07:02/0 5 0:00:00/0   any-call-type      0    0
1:07:96/0 0 0:00:00/0   voice-call-type    0    0
1:07:96/0 1 0:00:00/0   digital-call-type  0    0
1:07:96/0 2 0:00:00/0   phs-call-type      0    0
1:07:96/0 3 0:00:00/0   voip-call-type     0    0
1:07:96/0 4 0:00:00/0   v110-call-type     0    0
```

Rejecting collect calls on Brazilian R2 signaling lines

Brazilian ISPs can now accept or deny collect calls on lines configured with R2 signaling. By specifying a value for the `group-b-collect-signal` parameter, you can configure the MAX TNT to accept or reject collect calls.

Group-B-Collect-Signal

Description: For Brazilian R2 signaling lines, specifies the group-B-signal that the MAX TNT unit sends in response to a collect call.

Usage: Specify one of the following signal values:

Signal-B-2—Indicates a busy line.

Signal-B-5—Indicates a line for which there is no fee.

Signal-B-7—Indicates that the line does not accept collect calls. Also indicates a number that is not accessible or that the call is forwarded to an answering machine.

Dependencies: If the `signaling-mode` parameter is set to any value other than `e1-brazil-signaling`, then the `group-b-collect-signal` parameter does not apply.

Example: `set group-b-collect-signal = signal-b-7`

SS7 extension features

Q.931+ for PacketStar SS7 signaling gateways

Q.931 is an ISDN connection control protocol defined in ITU-T specifications. It performs control signaling on the ISDN D channel to manage connection setup and breakdown on 64-Kbps B channels. Q.931 is designed to provide access between a user side and network side, and its messages are related to call control (for example, CONNECT and DISCONNECT).

Q.931+ is based on Q.931 but provides additional functions required for the SS7 signaling gateway interface. With reference to the Q.931 specification, the signaling gateway implements the network side and the MAX TNT unit (the access server) implements the user side of the Q.931 protocol.

The MAX TNT unit's trunk interfaces are connected to the remote switch via intermachine trunks (IMTs). The channels of the trunk lines are identified by an interface number and a channel number, in a manner similar to non-facility associated signaling (NFAS) Q.931 signaling. An incoming modem or HDLC data call can be connected to any available channel in the unit.

A mapping between the circuit identification code (CIC) at the switch and the combined interface identifier and channel number in the MAX TNT trunk interfaces is configured and translated at the signaling gateway.

In addition to the Q.931+ control protocol, Lucent signaling gateway platforms support the protocols IP Device Control (IPDC) 0.12 and Access SS7 Gateway Control Protocol (ASGCP)-Q.931+. Table 18 shows the signaling protocols supported by signaling gateway platforms that interoperate with the MAX TNT for SS7 support.

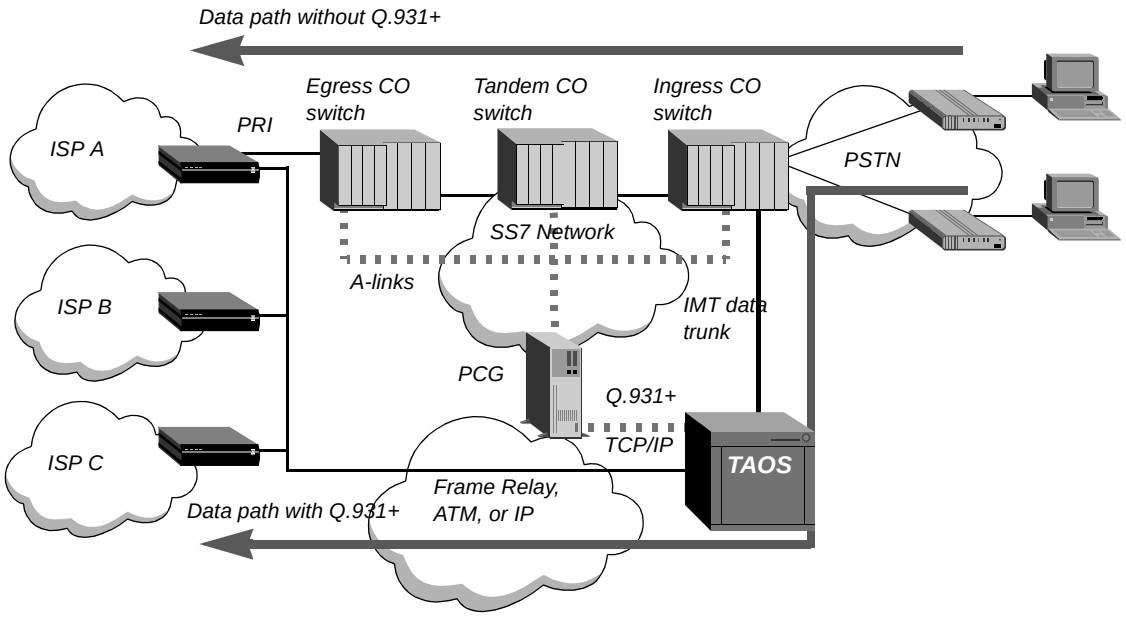
Table 18. Signaling gateway platforms and protocol support

Platform	**IPDC 0.12**	**Q.931+**	**ASGCP-Q.931+**
ICD for softswitch (formerly ASG)	Supported	Not supported	Supported
Lucent Technologies Softswitch	Supported	Not supported	Not supported
PacketStar™ Connection Gateway (PCG)	Not supported	Supported	Not supported

Terminating data calls in an SS7 network

With the `q.931-plus` license, MAX TNT units can decrease congestion on the Public Switched Telephone Network (PSTN) caused by users connecting to the Internet. An example of MAX TNT units being used for this purpose is shown in Figure 2.

Figure 2. MAX TNT unit terminating data calls in an SS7 network



The MAX TNT unit is connected to the entry (ingress) central office (CO) switch via intermachine trunks (IMTs) and to a signaling gateway by means of dual-link (primary and secondary) TCP/IP links. Each CO switch is a service switching point (SSP). The combination of a MAX TNT unit and signaling gateway is also an SSP. The signaling gateway is connected to the SS7 network by access links (A-links). The signaling gateway and the MAX TNT unit together act as a switch that routes calls intended for ISPs directly to the MAX TNT unit, thus avoiding the PSTN tandem or transit switches and interoffice trunks.

Simple data delivery layer (DDL)

The simple data delivery layer (DDL) uses TCP/IP for reliable transmission between the signaling gateway and MAX TNT unit.

The DDL prefers the primary link over the secondary and always tries to bring up the primary link. It uses a 4-second keep alive mechanism to verify that the link is up. If the primary link is inactive, the DDL uses the secondary link. If the primary link subsequently becomes active, the DDL switches back to the primary.

The DDL uses a 2-octet header to encapsulate the control protocol messages. The header contains the length of the SS7 message.

Overview of configuration settings

When the `q.931-plus` license is enabled, most of the SS7-Gateway profile settings operate as documented for other control protocols. For details about the following parameters, see the *APX 8000/MAX TNT/DSLTNT Reference*.

```
[in SS7-GATEWAY]
```

```
enabled = no
control-protocol = asgcp
primary-ip-address = 0.0.0.0
primary-tcp-port = 0
```

```
secondary-ip-address = 0.0.0.0
secondary-tcp-port = 0
bay-id = ""
system-type = IASCTNT1B
transport-options = { ascend 0 1000 3000 30000 7 6 no }
use-system-ip-address-as-source = yes
congestion-control = { l3-queue-depth 60 send-info-to-mgc 120 reject-
new-call }
signaling-heartbeat = { no 3 }

[SS7-GATEWAY:transport-options]
heartbeat = no
```

Following are the parameters that are specific to a Q.931+ configuration. The parameters are shown with the default settings when the `q.931-plus` license is the only licensed control protocol.

```
[in SS7-GATEWAY]
control-protocol = q.931-plus

[SS7-GATEWAY:transport-options]
type = ascend

[in T1/{ any-shelf any-slot 0 }:line-interface]
nfas-id = 0

[in E1/{ any-shelf any-slot 0 }:line-interface]
nfas-id = 0
```

Parameter	Specifies
Control-Protocol	Control protocol. Valid values are `asgcp` (for communicating with an ICD for softswitch using ASGCP-Q.931+), `ipdc-0.x` (for communicating with a Lucent Technologies Softswitch using IPDC), or `q931-plus` (for communicating with a PacketStar Connection Gateway). If only one control protocol is licensed, the setting defaults to the licensed protocol and cannot be modified. However, if multiple protocols are licensed, the parameter defaults to `asgcp`. Because of this default and because the MAX TNT unit does not store unmodified profile items in NVRAM, the setting can be modified unintentionally when you upgrade to new software or enable a new license to support another SS7 control protocol. For this reason, Lucent recommends that you verify the setting after upgrading. If the proper protocol is not specified, change the setting and then reset the unit. Although the control protocol is configurable in real time, you must reset the system to begin using the new protocol. After the reset, the unit establishes a new TCP link to the signaling gateway and begins communicating with it using the specified protocol.

Parameter	Specifies
Type	Type of transport layer. The default `ascend` setting indicates the transport layer used by the ASGCP and IPDC protocols: TCP/IP-based data delivery with built-in redundancy and retransmission. If the parameter is set to `tcp-encaps-2`, the system uses a TCP/IP stream with a 2-octet header added to every signaling message. The `tcp-encaps-2` setting is required for Q.931+ and does not apply unless Control-Protocol is set to `q931-plus`. **Note:** When Type is set to `tcp-encaps-2`, all parameters in the Transport-Options subprofile except the Heartbeat parameter are not applicable.
NFAS-ID	An interface ID for the T1 or E1 line used as an SS7 line. This setting applies to SS7 lines only when the Q.931+ control protocol is used.

Support for Q.931+ status messages

The MAX TNT unit supports the following messages about changes in status related to Q.931+:

Messages	Status changes
NAS_STATUS and NAS_STATUS_ACK	Device registration on cold and warm startup. A cold startup is a system reset. A warm startup is a signaling link reestablishment or failover.
INTERFACE_STATUS and INTERFACE_STATUS_ACK	Registration of all SS7 line interfaces upon establishment of the signaling link and notifications of any status or configuration changes thereafter.
RESOURCE_UPDATE and RESOURCE_UPDATE_ACK	Registration of all available host resources (HDLC and modem) upon establishment of the signaling link and notifications of any status or capacity changes thereafter.

A NAS_STATUS message is sent whenever the TCP/IP transport link between the MAX TNT unit and the signaling gateway comes up. The status of the device is reported as *cold start* the first time the signaling connection is established. After that, failovers and reestablishment of the signaling link cause the system to send a NAS_STATUS message reporting its status as *warm start*. The status is reported as *cold start* if the SS7-Gateway profile is disabled and then enabled again.

A RESOURCE_UPDATE message that is sent to register host resources (modems and HDLC) always contains two “AS Resource” information elements. The first element in the message reports the capacity of the resource that has actually changed. (A change in resource capacity is the reason for RESOURCE_UPDATE to be generated by the MAX TNT unit.) The second information element in the message contains information about the other resource. (For example, if modem capacity changes, the MAX TNT unit sends a RESOURCE_UPDATE with two resources: the first is modem capacity and the second is HDLC capacity. If the HDLC capacity changes, the first resource is HDLC and the second is modem.)

SNMP support for Q.931+

The new value `q931plus` is supported for the `mgProtocol` object in `mgstat.mib`, as shown in the following definition:

```
mgProtocol  OBJECT-TYPE
SYNTAX      INTEGER {
            notApplicable(1), -- MG control is not
                              -- enabled by hash code, etc.
            other(2),    -- Other (none from the list below)
            asgcp(3),    -- ASGCP (Ascend Signaling Gateway
                              -- Control Protocol)
            ipdc(4),     -- IPDC (Internet Protocol Device
                              -- Control)
            q931plus(5) -- Q.931+ (Q.931 with extensions
                              -- for IMT signaling over IP)
            }
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "Type of the control protocol in use."
::= { mgTableEntry 2 }
```

Log message support for Q.931+

The MAX TNT unit now reports several new messages related to Q.931+ call processing. It reports the messages to its logging facilities (Log profile and Syslog) according to its configuration. For details about log facilities, see the *APX 8000/MAX TNT/DSLTNT Administration Guide*.

Error messages related to Q.931+

The following messages are logged at the ERROR level:

`LOG error, Shelf 1, Slot 17, Time: 18:45:41--`
`Q.931+ nfas-id=`*`N`* `is invalid. Check line config for {`*`X Y Z`*`}.`

`LOG error, Shelf 1, Slot 17, Time: 18:45:41--`
`Q.931+ nfas-id=`*`N`* `is not unique. Check line config`
`for {`*`X Y Z`*`} and {`*`A B C`*`}.`

In the preceding messages, the following definitions apply:

- *N* is a value assigned to the NFAS-ID parameter that is invalid according to the Q.931+ specification or is not unique in the system.
- `{X Y Z}` is the physical address of the line (the index of the T1 or E1 profile) in which the invalid or duplicate NFAS ID was detected.
- `{A B C}` is the physical address of the line (the index of the T1 or E1 profile) in which the duplicate NFAS ID was first detected.

Notice messages related to Q.931+

The following messages are logged at the NOTICE level and reflect authorization level received by the MAX TNT unit during Q.931+ signaling link establishment:

`LOG notice, Shelf 1, Slot 17, Time: 21:48:46--`
`Q.931+: Registered for incoming calls only. Cause=`*`DDD`*`.`

```
LOG notice, Shelf 1, Slot 17, Time: 21:48:46--
Q.931+: Registered for incoming and outgoing calls. Cause=DDD.

LOG notice, Shelf 1, Slot 17, Time: 21:48:46--
Q.931+: Access server registration rejected by SG. Cause=DDD.

LOG notice, Shelf 1, Slot 17, Time: 21:48:46--
Q.931+: Access server registration failed. Cause=DDD.
```

In the preceding messages, *DDD* is the decimal value of the cause code received by the MAX TNT unit in the Q.931+ NAS_STATUS_ACK message.

Enhancement to Q.931+ debug tracing capability

Q.931+ tracing capability is enhanced with the addition of the `dump decoded IEs` debug level (level 4) to the diagnostics option of the `diag ss7asg 0x04` command. When debug level is set to 4, the following information elements (IEs) are decoded and displayed in real time:

- Bearer capability
- Called party number
- Calling party number
- Cause value
- Call reference
- Channel identification

Where previously information elements were displayed in hexadecimal format, decoded information elements now take the following form:

```
RECV:SETUP
BEARER CAPABILITY :
  INFO XFER CAPABILITY :Speech
  USER LAYER 1 PROTO :User Layer 1 None
CALLED PARTY NUMBER:
  TYPE OF NUMBER: National
  NUMBERING PLAN:ISDN
  CALLED PARTY DIGITS:6501234567
CALLING PARTY NUMBER:
  TYPE OF NUMBER: International
  NUMBERING PLAN:ISDN
  PRESENTATION INDICATION:Presentation allowed
  SCREENING INDICATION:User-provided, not screened
  CALLING PARTY DIGITS:5107471234
RECV:DISCONNECT
CAUSE VALUE:
CODING STANDARD:Network Specific
LOCATION :Private network serving local user
CAUSE :16
RECV:RELEASE COMPLETE
CAUSE VALUE:
CODING STANDARD:Network Specific
LOCATION :Private network serving local user
CAUSE :16
```

In addition to the display of decoded information elements, the call-processing message type transmitted or received between the signaling gateway and the MAX TNT unit is also displayed.

The interface is modified as follows:

```
admin> ss7asg
        usage:ss7asg -option [ params ]
                -i              Show interfaceID map
                -m              Display all MCBs (ME control blocks)
                -n              Display all NLCBs (L3 call blocks)
                -r              Reset Signaling Layer  statistics
                -s              Show SS7 interface statistics
```

To enable diagnostics at a specific level, enter the following command:

```
admin> diag ss7asg level
```

Level	Description
0x00	Disables diagnostic output
0x01	Shows errors only
0x02	Traces L3 events and states
0x04	Traces call control events
0x08	Dumps decoded IEs
0x10	Shows all task events
0x20	Enables code trace for debugging
0x40	Dumps L3 packets
0x80	Dumps call control primitives
0x100	Debugs signaling link
0x200	Shows memory allocation and deallocations

Note: Existing debug levels are reordered.

SS7 Q.931 messaging support for V.110 calls

MAX TNT units can now recognize and respond to PacketStar Connection Gateway (PCG) call setup requests for V.110 processing of Signaling System 7 (SS7) calls through the Q.931 protocol. Q.931 requires V.110 bearer capability at certain unrestricted adaptation rates now supported by TAOS. This release adds a second octet, octet 5a, to the information element (IE) of a Q.93 call setup message sent by a PacketStar Connection Gateway (PCG).

Supported Q.931 bearer capability requests

For calls requiring V.110 bearer capability, a PacketStar Connection Gateway generates one of the following Q.931 call setup message requesting one of the following adaptation rate ranges. The call type is set in octet 5 of the Bearer Cap information element of the message, and the adaptation rate is retrieved from the user rate in the same octet.

Request	Specifies
0x04 0x03 0x80 0x90 0xa0	Speech bearer capability.
0x04 0x04 0x88 0x90 0x21 0xc3	V.110 bearer capability with 2400bps to 64Kbps unrestricted adaptation.
0x04 0x04 0x88 0x90 0x21 0xc5	V.110 bearer capability with 4800bps to 64Kbps unrestricted adaptation.
0x04 0x04 0x88 0x90 0x21 0xc8	V.110 bearer capability with 9600bps to 64Kbps unrestricted adaptation.
0x04 0x04 0x88 0x90 0x21 0xcb	V.110 bearer capability with 19200bps to 64Kbps unrestricted adaptation.
0x04 0x04 0x88 0x90 0x21 0xcd	V.110 bearer capability with 38400bps to 64Kbps unrestricted adaptation.

Feature description

For calls requiring V.110 bearer capability, the PCG generates a Q.931 call set-up message requesting that bearer capability at one of the following unrestricted adaptation rates supported by MAX TNT units:

- 2400bps to 64Kbps
- 4800bps to 64Kbps
- 9600bps to 64Kbps
- 19200bps to 64Kbps
- 38400bps to 64Kbps

Octet 5a information element

When the Q.931 call setup message sent by the PacketStar Connection Gateway requests V.110 bearer capability, the following values must be assigned to octet 5a to enable Asynchronous Transfer Mode (ATM) and disable inband call signaling on a MAX TNT unit for the duration of the SS7 call:

Bit #	Value	Description
Bit 7	1	Enable asynchronous data mode for this call.
Bit 6	0	Disable inband negotiation.

SS7 IPDC RTE for congestion control (Lucent Technologies Softswitch only)

With TAOS 8.0.3, units with the SS7 IPDC license perform congestion control functions using Request Test Echo (RTE) messages with a congestion indicator and Acknowledge RTE (ARTE) response from the signaling gateway. If the signaling gateway is a Lucent Technologies Softswitch, it can respond to the congestion indication by slowing down the rate at which it is forwarding calls. For any other type of signaling gateway, the RTE messages are used as a signaling heartbeat.

Note: With TAOS 8.0.3, congestion control applies only to units that communicate with the Lucent Technologies Softswitch signaling gateway using IPDC. The ICD for softswitch signaling gateway will support congestion control in an upcoming softswitch release. This feature does not apply to units using the SS7 ASGCP license.

Using RTE messages as a signaling heartbeat

With TAOS 9.0, the default settings in the SS7-Gateway profile cause the unit to transmit RTE messages to the signaling gateway every 3 seconds. If congestion control is disabled, or if the signaling gateway does not support congestion control, the packets are used as a signaling heartbeat.

Format of RTE messages without congestion indicator

Following is the format of the RTE message without the congestion indicator:

```
Protocol ID: 0x4B
Transaction ID: 0x8xxxxxxx
Message: 0x007D
```

For example:

```
Protocol=0x4b, Correlator (4): 80024b13
Message: 0x007d
End of NMI message.
```

Overview of SS7-Gateway settings

Following are the relevant parameters, shown with default settings:

```
[in SS7-GATEWAY:signaling-heartbeat]
enabled = no
interval = 3
```

Parameter	Specifies
Enabled	Enable/disable signaling layer heartbeat to the signaling gateway. If set to `yes`, the unit sends a signaling heartbeat to the gateway at the interval specified in the Interval parameter. By default, this parameter is set to `no`.
Interval	Number of seconds between signaling heartbeat messages. The valid range is from 0 to 86400 (default 3).

Example of changing the heartbeat interval

The following commands configure the unit to sent RTE messages to the signaling gateway every second:

```
admin> read ss7-gateway
SS7-GATEWAY read

admin> set signaling-heartbeat interval = 1

admin> write
SS7-GATEWAY written
```

Example of disabling transmission of RTE messages

The following commands disable transmission of RTE messages altogether:

```
admin> read ss7-gateway
SS7-GATEWAY read

admin> set signaling-heartbeat enable = yes

admin> write
SS7-GATEWAY written
```

Using RTE messages for congestion control

By default congestion control is disabled. When congestion control is enabled, MAX TNT units monitor the depth of the layer-3 queue as a measure of call congestion. The queue contains messages for the IPDC layer, including call control and other network messages, as well as messages from IPDC itself.

When the number of messages in the queue exceeds congestion level 1, the unit can either ignore the congestion level or send an RTE message with a congestion level indicator that level 1 has been exceeded (the default). When the number of messages drops below the specified congestion level 1, the unit sends an RTE message indicating congestion level 0 (no congestion).

When the number of messages in the queue exceeds congestion level 2, the unit can either ignore the congestion, send an RTE message to the signaling gateway indicating that congestion level 2 has been exceeded, or send the message and reject new calls (the default).

Format of RTE messages with congestion indicator

Following is the format of the RTE message with the congestion indicator:

```
Protocol ID: 0x4B
Transaction ID: 0x8xxxxxxx
Message code: 0x007D
Tag ID = 0xa6, <congestion-level>
```

For example:

```
Protocol=0x4b, Correlator (4): 80024b13
Message: 0x007d
Tag ID = 0xa6, Data (1): 01
End of NMI message.
```

Tag `0xA6` can specify the following values:

Size	Type	Description	Value	Meaning	Usage
1	UINT	Congestion level indicator	`0x00`	Not congested	RTE/ARTE
			`0x01`	Congestion level 1	
			`0x02`	Congestion level 2	

Overview of SS7-Gateway settings

In addition to the signaling heartbeat parameters described in “Using RTE messages as a signaling heartbeat” on page 93, which control transmission of RTE messages, the following parameters (shown with default settings) are used for configuring congestion control:

```
[in SS7-GATEWAY:congestion-control]
congestion-control-type = l3-queue-depth
cl1-level = 60
cl1-action = send-info-to-mgc
cl2-level = 120
cl2-action = reject-new-call
```

Parameter	Specifies
Congestion-Control-Type	Congestion control algorithm to use. If set to `l3-queue-depth` (the default), the unit measures the depth of the layer-3 queue as the criterion of congestion. If set to `none`, congestion control is disabled.
CL1-Level	Number of messages in the queue (from 0 to 1000) at which the unit informs the signaling gateway that congestion level 1 has been exceeded. By default, congestion level 1 occurs when the queue contains 60 messages.
CL1-Action	Action to perform when congestion level 1 (defined by the value of the CL1-Level parameter) has been exceeded. Valid values are `send-info-to-mgc` (the default) and `ignore`. If set to `send-info-to-mgc`, the unit sends an RTE message to the signaling gateway with the appropriate congestion level indicator. If set to `none`, the unit takes no action at congestion level 1.
CL2-Level	Number of messages in the queue (from 0 to 1000) at which the unit informs the signaling gateway that congestion level 2 has been exceeded. By default, congestion level 2 occurs when the queue contains 120 messages.
CL2-Action	Action to perform when congestion level 2 (defined by the value of the CL2-Level parameter) has been exceeded. Valid values are `reject-new-call` (the default), `send-info-to-mgc`, and `ignore`. If set to `reject-new-call`, the unit rejects new calls and sends an RTE message to the signaling gateway with the appropriate congestion level indicator. If set to `send-info-to-mgc`, the unit sends an RTE message to the signaling gateway with the appropriate congestion level indicator. If set to `ignore`, the unit takes no action at congestion level 2. If you set CL1-Level and CL2-Level to a low value and set CL2-Action to `reject-new-calls`, the unit begins rejecting calls before it reaches its maximum call-processing capacity. If you set CL1-Level and CL2-Level to a high value and set CL2-Action to `reject-new-calls`, the unit attempts to use all of its call-processing capacity, even though the calls might time out at the signaling gateway. (The Lucent Technologies Softswitch timeout is configurable with a default of 3 seconds.)

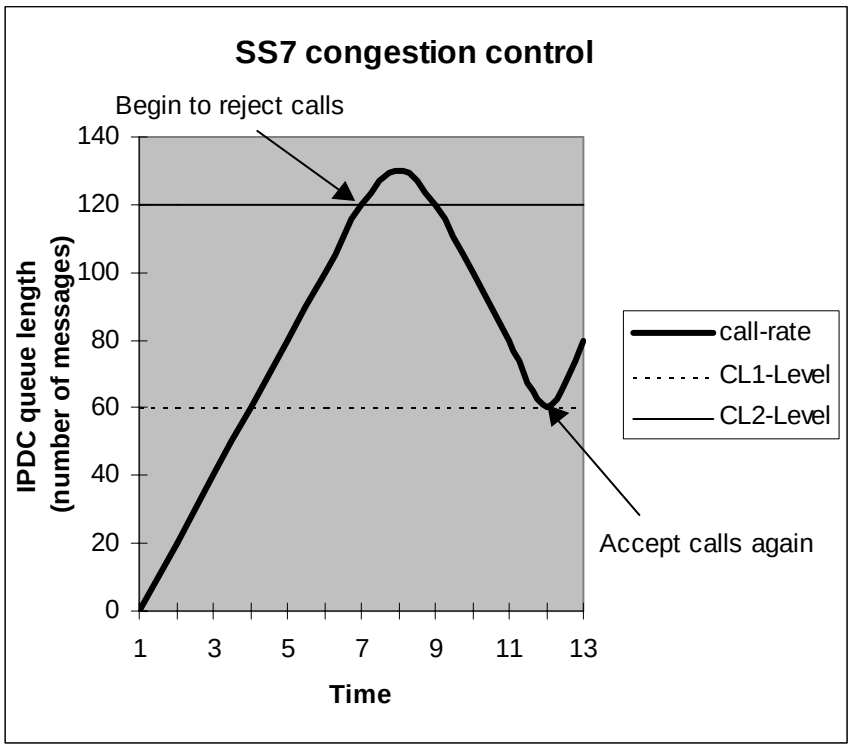
Example of configuring congestion indication

The commands in the following example configure the MAX TNT unit to send a level-1 congestion indicator to the signaling gateway when the layer-3 queue contains 100 messages, and to reject new calls and send a level-2 congestion indicator when the queue contains 200 messages. In addition, the commands specify that the RTE exchange must occur every 10 seconds.

```
admin> read ss7-gateway
SS7-GATEWAY read
admin> set congestion-control cl1-level = 60
admin> set congestion-control cl2-level = 120
admin> set signaling-heartbeat enabled = y
admin> set signaling-heartbeat interval = 10
admin> write
SS7-GATEWAY written
```

Figure 3 shows how the MAX TNT unit rejects and accepts calls over a period of time on the basis of the settings in the preceding configuration example.

Figure 3. Results of SS7 congestion control



SS7 IPDC: Reporting VoIP call statistics

In this release, the MAX TNT unit operating as network access server (NAS) reports VoIP call statistics when a call is cleared, whether the call clearing is initiated by the signaling gateway or the MAX TNT. The statistics are reported via IPDC when the statistics are available.

Supported statistics tags (IPDC 0.12)

The following table shows statistics-related tags from IPDC 0.12 that are currently supported by the MAX TNT, and describes how Lucent interprets those tags:

Tag	Description
`0x91`	Number of packets sent and received. Lucent interprets this as the number of Real-Time Transport Protocol (RTP) audio packets sent by the MAX TNT.
`0x92`	Number of packets dropped. Lucent interprets this as the number of RTP audio packets that failed to reach the MAX TNT, as determined by missed sequence numbers.
`0x93`	Number of bytes sent and received. Lucent interprets this as the number of audio RTP payload bytes sent by the MAX TNT.
`0x94`	Number of bytes dropped. Lucent interprets this as the number of audio RTP payload bytes that failed to reach the MAX TNT. Because the number of bytes per packet varies, this value can only be estimated, based upon an average packet size multiplied by the number of nonreceived packets. This value can also be estimated by the control server with the information supplied.
`0x9D`	Number of audio packets received. Lucent interprets this as the number of RTP audio packets received.
`0x9E`	Number of audio bytes received. Lucent interprets this as the number of audio RTP payload bytes received.
`0xA3`	Estimated interarrival jitter in milliseconds. A new tag implemented by Lucent. 0xA3 is the next available tag value in IPDC 0.15. Estimated interarrival jitter is computed as follows: `J = J + (D - J)/16` where `D = | R(i) - R(i-1) - T |`, `R(i)` is the arrival time of the received packet `i`, and `T` is the theoretical difference of departure time between two consecutive packets at the source. For example, `T` is 5 ms for G711 1 frame per packet, `T` is 10 ms for G729 1 frame per packet, and `T` is 40 ms for G729 4 frames per packet.

Unsupported statistics tags (IPDC 0.12)

The following table shows statistics-related tags from IPDC 0.12 that are not currently supported by the MAX TNT:

Tag	Description
0x95	Number of signaling packets sent and received.
0x96	Number of signaling packets dropped.
0x97	Number of signaling bytes sent and received.
0x98	Number of signaling bytes dropped.
0x99	Estimated average latency.
0x9F	Number of signaling packets received.
0xA0	Number of signaling bytes received.

Call statistics reporting

IPDC 0.12 specifies that the statistics tags are optional, and they are reported in the following cases of call clearing:

- When the access server initiates a call teardown via a release connection request (RCR) message.
- For packet-based calls when the access server acknowledges a call teardown via a release connection completed (ACR) message.

The MAX TNT reports the statistics in the above two cases when the statistics are available.

SS7 NMI command enhancements

The `ss7nmi -s` command has been enhanced to display the number of RCR and ACR messages sent with or without VoIP call statistics and the number of unknown messages from VoIP control bus to the SS7 module. The new statistics are displayed in **bold** in the following sample output:

```
admin> ss7nmi -s
SS7 NAS Messaging Interface (NMI) statistics
        Initialized successfully                  Yes
        Total number of internal errors           0
        Level of diagnostics                      1
        Resource backtrace collection             disabled
Signaling Layer
        Current link state                        UP
        Last generated transaction ID             19
        Timer Trst1                               1000  ticks - idle
        Timer Tnsup                               1000  ticks - idle
        Number of protocol version errors         0
        Number of 'message reject' received       0
        Number of bad packets received            0
        Number of unknown messages                0
        Number of unknown SS7Voip messages        0
```

```
        Number of resource conflicts              0
        Number of release race conditions         0
        Number of RCR with stats sent             0
        Number of RCR without stats sent          0
        Number of ACR with stats sent             0
        Number of ACR without stats sent          1
        Number of busy reject                     0
        Number of IPDC queue congestion reject    0
        Number of RTE timeout                     0
Data Transport Layer
        Number of link fail-overs                 0
        Number of persistent errors               0
        Last error                                No Error
        Last error timestamp                      [01/01/1990 000000]
```

SS7: PRI tunneling in IPDC (IPDC 0.15)

TAOS 9.0 adds a new PRI signaling type that allows ISDN Layer 3 signaling to be tunneled to an external signaling gateway. This feature makes the PRI lines terminating on MAX TNT unit controllable by an external signaling gateway.

This enhancement makes ISDN PRI lines terminating on a MAX TNT unit deployed as an access gateway or as a trunking gateway in VoIP networks visible to an external signaling gateway using IP Device Control (IPDC) protocol to perform call control on this lines. In this tunneled PRI signaling scheme, a MAX TNT unit handles layer 2 and layer 1 PRI signaling. All layer 3 Q.931 messages on the D-channel are tunneled to the Media Gateway Controller (Softswitch) using the IPDC TUNL message. The bearer channels on the PRI lines will be controlled by IPDC call setup and teardown messages.

This feature requires a MAX TNT unit be hashed for IPDC. The current release supports only ISDN network terminated (NT) emulation for T1 and T3 lines connected to NI-2 and 5ESS/4ESS ISDN switch types.

Note: Only one signaling type can be used on a MAX TNT unit's channelized T1 card.

To support SS7 PRI tunneling, a MAX TNT unit requires the following:

- IPDC signaling must be enabled on the MAX TNT. This may be verified by checking the Base profile for the `xcom-ss7=enabled` entry.
- IP address and TCP port to use as the IPDC interface to the SS7 signaling gateway. Typically, the primary and secondary address and port configurations point to the two Ethernet interfaces of the SS7 signaling gateway. This is configured by assigning the appropriate IP address to the Primary-Ip-Address parameter and the appropriate port number to the Primary-Tcp-Port parameter in the SS7-Gateway profile.

User interface changes

The enhancement adds a new value to the Signaling-Mode parameter in the T1 Line Interface profile that allows an external signaling gateway using IPDC to perform call control on T1 lines terminating on a TAOS unit. It also creates a new debugging command for testing

tunnelled signaling operations, and modifies the line status output to report activity on DS0 channels using tunnelled signaling.

Signaling-Mode

Description: This parameter is modified to enable ISDN Layer 3 signaling to be tunneled to an external signaling gateway. When this signaling type is enabled, all layer 3 Q.931 messages are tunneled to the gateway configured in the SS7-Gateway profile.

Usage: Setting the value of the Signaling-Mode parameter to `tunneled-pri-signaling` value enables the MAX TNT unit to recognize and respond to the ISDN signaling, with local B-channels controlled by an external Media Gateway Controller . Once selected, PRI tunneling is enabled with the next VoIP call.

Dependencies: When `signaling-mode=tunneled-pri-signaling`, PRI tunneling for SS7 VoIP calls is only supported when IPDC signal processing is enabled for this MAX TNT unit. The Base profile should contain the following entry:

```
xcom-ss7=enabled
```

Location: T1 { x x x } > Line-Interface

Reporting PRI tunneling status

A new symbol, "`i`", is used by the Status command to report an active tunneled PRI trunk. This symbol identifies DS0 connections that use ISDN PRI with layer 3 tunneled signaling to an external signaling gateway, as illustrated by the following:

```
0 Connections, 0 Sessions| TNT22 Status
                         | Serial number: 9021340  Version: 9.0a0e0
                         |
                         |  Rx Pkt: 27763
                         |  Tx Pkt: 14688
                         |     col:     2
                         |
                         | 04/06/2000 18:29:42  Up: 0 days, 02:30:20
                         |-----------------------------------------
                         |T-PRI 1/01/01 LA i-------- -------- -------s
                                            ^
```

Using the Tunlpri command

The Tunlpri command is used to report the status of calls processed using tunneled PRI signaling. This command uses the following syntax:

```
tunlpri -s
```

Using the `-s` option, the Tunlpri command will display module statistics for T1 and T3 connections. To enable tunneled PRI diagnostics, use the following Diag command to set the desired level for debugging tunneled PRI operations:

```
diag tunlpri <level>
```

Debug level	Specifies
`0x00`	Diagnostic output is disabled. No debugging information is collected.
`0x01`	Report errors only. Collect only high level error information as errors occur.

Debug level	Specifies
0x02	Show basic debugging traces. Collect session logs.
0x04	Dump Tunnel messages. Collect the IPDC TUNL messages sent to and received form the Softswitch.
0x08	Show detailed debugging traces. Collect full session logs, including low-level processing information for tunneled PRI signaling.

The following example illustrates the output of the Tunlpri using debug level four (0x04):

```
admin> tunlpri  -s
   Tunneled PRI Module statistics:
           Interface initialized and ready:            Yes
           Current level of diagnostics:               15
   Message count:
           Received from L2  :                         1068
           Sent to L2        :                         754
           Received from Tunl:                         945
           Sent to Tunl      :                         996
   Errors:
           Errors at startup:                          0
           Warnings:                                   0
           Module usage errors:                        0
           NULL pointers:                              0
           Control Bus errors:                         72
           Buffer pools errors:                        0
           Protocol errors:                            0
           Total:                                      72
```

Modifications to the ss7nmi command

The `ss7nmi` debug-level command now reports TUNL message statistics when executed as follows:

```
ss7nmi -m
```

When the command is executed with the `-m` option, the results displayed include the number of tunneled PRI (TUNL) messages sent or received by the MAX TNT unit. TAOS 9.0 modifies the ss7nmi debug command to include revisions of the following options specifically for IPDC Tunneling debugging:

Options	Specifies
`-m`	Show TUNL message statistics statistics.
`-mr`	Reset TUNL message statistics.
`-n`	Show active NLCBs (transactions).
`-r [address]`	Show the status of the SS7 circuit(s). When address is specified, show only status for the selected circuit.
`-rc`	Toggle, enable or disable, resource backtrace collection. By default, this option is disabled.
`-rd [address]`	Show detailed status of circuit(s). When address is specified, show only status for the selected circuit.
`-s`	Show SS7 interface statistics.

Options	Specifies
`-sr`	Reset SS7 interface statistics.

To enable tunneled PRI diagnostics, use the following Diag command to set the desired level for tracing tunneled PRI messaging:

```
diag ss7nmi <level>
```

Debug level	Specifies
`0x00`	Diagnostic output is disabled. No debugging information is collected.
`0x01`	Report errors only. Collect only high level error information as errors occur.
`0x02`	Show signaling link states. Collect information on SS7 link state changes.
`0x04`	Show NLCB/transaction states. Collect information on NCLB statuses and transactions state changes.
`0x08`	Show signaling semantics. Collect information on signaling types associated with each call.
`0x10`	Display contents of NMI packets. Collect information from network management information packets.
`0x20`	Show call control interface details. Collect information on the interface used to set up, monitor and tear down each call.
`0x40`	Show internal task events. Collect information on the low-level processes used for call control.
`0x80`	Show memory usage. Collect information on the memory allocated by the TAOS unit to process calls.
`0x100`	Show resource allocation details. Collect information on how TAOS unit resources are allocated for each call.
`0x200`	Show tunnel basic errors. Collect only high level tunneling PRI error information as errors occur.
`0x400`	Show tunnel basic debug. Collect only high level tunneling PRI debugging information for calls as they occur.
`0x800`	Dump Tunnel messages. Collect the IPDC TUNL messages sent to and received form the Softswitch.
`0x1000`	Show detailed debugging traces. Collect full session logs, including low-level processing information for tunneled PRI signaling.

The following example illustrates the output of the ss7nmi -m command, reporting the TUNL messaging statistics:

```
admin> ss7nmi -m
IPDC message processing statistics:
        Message code      Received       Sent
        RCR   (0x0011):     152802          0
        ACR   (0x0012):          0     152802
        RCCP  (0x0013):     152847          0
```

```
ACCP  (0x0014):          0     152847
RMS   (0x0041):          1          0
NMS   (0x0042):          0         24
RLS   (0x0043):         28          0
NLS   (0x0044):          0         29
NCS   (0x0046):          0          7
TUNL  (0x007a):     611460     611480
RTE   (0x007d):        111          0
ARTE  (0x007e):          0        111
NSUP  (0x0081):          0          1
ASUP  (0x0082):          1          0
Data collection was started: [04/08/2000 17:24:01]
```

SS7 gateway IPDC support for E1 trunks

TAOS 9.0 now supports SS7 gateway IP Device Control (IPDC) on E1 trunks on MAX TNT units.

Prior to this release, TAOS supported Ascend Signaling Gateway Control Protocol (ASGCP) on E1 and T1 trunks, whereas IPDC supported only T1 trunks. Now IPDC provides support on both E1 and T1 trunks. In contrast to ASGCP support, IPDC provides a more robust set of features that offer direct control over Voice-over-IP (VoIP) applications and SS7 circuits, such as inband tone generation and detection, as well as generation of announcements and test tones. IPDC can be used in a mixed environment that is supported by a Softswitch Media Gateway Controller.

IPDC setting added to the Control-Protocol parameter

To configure IPDC support from the command-line interface (CLI) of a MAX TNT unit, you must set the Control-Protocol parameter in the SS7-Gateway profile to `ipdc-0.X`.

Parameter	Specifies
Control-Protocol	Signaling protocol that controls the SS7 gateway. Following are valid values: `asgcp`—Sets the signaling gateway control to the proprietary ASGCP. `ipdc-0.X`—Sets the signaling gateway control to IPDC support for SS7 gateways. **Note:** The `ipdc-0.X` option offers XCOM/Level 3 IPDC 0.12. `q931-plus`—Sets the signaling gateway control to IMT support over IP Q.931.

Activating IPDC on an SS7 gateway

From the SS7-Gateway profile, you must set the Control-Protocol parameter to `ipdc-0.X` as follows to enable IPDC support:

```
admin> set control-protocol = ipdc-0.X
```

The following sample SS7-Gateway profile provides a configuration for IPDC support:

```
[in SS7-GATEWAY]
enabled = yes
control-protocol = ipdc-0.X
primary-ip-address = 192.168.1.1
primary-tcp-port = 9000
secondary-ip-address = 0.0.0.0
secondary-tcp-port = 0
bay-id = ""
system-type = IASCTNT1B
transport-options = { ascend 0 1000 3000 30000 7 6 yes }
use-system-ip-address-as-source = yes
congestion-control = { l3-queue-depth 60 send-info-to-mgc 120
reject-new-call }
signaling-heartbeat = { no 3 }
```

Note: You must reset the MAX TNT unit when switching configuration options from ASGCP to IPDC support. Otherwise, first-time activation of either feature is configured in real time.

SS7 continuity checks for E1 lines

In this release, the MAX TNT unit implements a Signaling System 7 (SS7) 4-wire and 2-wire continuity check for E1 as defined in ITU Recommendation Q.724, sections 7 and 8.

Overview of E1 line continuity checks

A continuity check can be performed at the time of call setup or during testing to verify that the physical link between the central office (CO) switch and the MAX TNT unit is available. The CO switch informs the signaling gateway, which then informs the MAX TNT unit that it will conduct a continuity test on the circuit. During a call continuity test, the CO switch sends a tone through the physical path to the MAX TNT unit and receives a tone back from the MAX TNT unit indicating the continuity of the path.

With this release, the MAX TNT unit supports incoming and outgoing 2-wire and 4-wire continuity checks for E1 lines. You can select the type of check to perform on a per-line basis. Both the native 2-wire continuity check (GR-246-CORE Section B.2) and 4-wire-to-2-wire emulation (GR-246-CORE Section B.3) are supported.

A 4-wire continuity check requires one end of a line to place a channel into loopback state while the other end sends a tone. The check concludes successfully if the tone sent on the outgoing path is received on the return path within acceptable transmission and timing limits. The 4-wire check procedure cannot detect potential inadvertent loops in the line path or in line facilities, and cannot be used when the other exchange is analog. For these reasons, the procedure known as 2-wire continuity check is recommended by the International Telecommunications Union Telecommunication Standardization Sector (ITU-T).

Dependencies

The type of the continuity check procedure to be used is determined by line provisioning and is agreed upon by the connecting exchanges. SS7 signaling procedures used for continuity check (Q.764 Section G.3, ANSI T1.113.4 Section 2.1.6) are the same for both 4-wire and 2-wire circuits, but the behavior of trunk termination devices is different.

The native 2-wire continuity check procedure requires that the loopback be replaced by a transponder and that a 1780Hz ± 20Hz tone be used in the return direction.

The MAX TNT unit also supports the 4-wire-to-2-wire continuity check, with the following requirements: The exchange that terminates 4 wires must use a transmitting frequency of 1780 ± 20Hz and a receiving frequency of 2000 ± 20Hz. The exchange that terminates the 2 wires must use a transmitting frequency of 2000 ± 20Hz and a receiving frequency of 1780 ± 20Hz.

Configuring SS7 continuity checks for E1 lines

An SS7-Continuity subprofile has been added to the E1 profile to allow you to specify the type of incoming and outgoing continuity checks to perform for all channels on a line. Both ends of the connection must agree on the continuity check to be used for the line. Following are the relevant parameters, shown with default values:

```
[in E1/{ shelf-1 slot-1 1 }:line-interface:ss7-continuity]
incoming-procedure = loopback
outgoing-procedure = single-tone-2000
```

Parameter	Specifies
Incoming-Procedure	Loopback or transponder test mode. The `loopback` setting (the default) places the channel into loopback mode during the continuity test. This mode must be used if the line is provisioned for an incoming 4-wire continuity test. The `transponder` setting places the channel into Tone Transponder mode during the continuity test. In this mode, the channel can detect two tones: 2000Hz and 1780Hz. When either tone is detected, the other one is returned. This mode must be used for lines provisioned for incoming 2-wire and 4-wire-to-2-wire continuity checks.
Outgoing-Procedure	Type of continuity check. With the `single-tone-2000` setting (the default), the MAX TNT unit sends a 2000Hz tone and expects to receive a 2000Hz tone in return. This procedure is generally known as a 4-wire continuity check. With the `send-2000-expect-1780` setting, the MAX TNT unit sends a 2000Hz tone and expects to receive 1780Hz tone in return. This procedure is generally known as a 2-wire continuity check. With the `send-1780-expect-2000` setting, the MAX TNT unit sends a 1780Hz tone and expects to receive a 2000Hz tone in return. This procedure is generally known as a 4-wire to 2-wire continuity check. If you change the type of a continuity check, the new type is used for new continuity check requests on the line as soon as the line profile is saved. Existing check-loops that are already active on the line are not modified or canceled when the profile is saved.

MultiVoice extension features

The new features described in this section support MultiVoice functionality on the MAX TNT unit.

NavisAccess support for VoIP call reporting

This enhancement provides basic VoIP call reporting using NavisAccess. MAX TNT units, configured as MultiVoice gateways, respond to requests from NavisAccess for start records, stop records, and call progress records for both VoIP and fax calls. These records allow NavisAccess to monitor the resource usage of a MAX TNT unit running the MultiVoice gateway software, and provide information to create billing records. Each VoIP call can generate two or more records.

Start records

A start record reports the point in the call where a speech communications is established. Start records can provide the following information:

Attribute	**Specifies**
Ascend-Call-Direction	Direction of the call between the MultiVoice gateway and PSTN. The reported values are Ascend-Call-Direction-Incoming (0) and Ascend-Call-Direction-Outgoing (1). (Ascend Trap 48)
NAS-Port	Encoded network access server (NAS) port used for this call. (RFC Trap 5)
NAS-Port-Type	Encoded NAS port used for this call. The value 7 for this attribute identifies a VoIP call. (RFC Trap 61)
NAS-IP-Address	NAS IP address associated with this call. (RFC Trap 4)
Session-Id	NAS session index recorded in the session table for this call. (RFC Trap 44)
Ascend-Modem-PortNo	DSP or modem port allocated for processing this call. This value is part of the resource count information and is repeated each time it is allocated for a call. (Ascend Trap 120)
Ascend-Modem-SlotNo	Slot where the DSP or modem card associated with the reported Ascend-Modem-PortNo is located. This value is part of the resource count information, and is repeated each time it is allocated for a call. (Ascend Trap 121)
Ascend-Modem-ShelfNo	Shelf where the controller for the DSP or modem card allocated for processing this call is installed. This is part of the resource count information, and is repeated each time it is allocated for a call. (Ascend Trap 122)
Called-Station-Id (DNIS)	Dialed number string reported by the MultiVoice gateway for the called destination. (RFC Trap 30)
Ascend-Dialed-Number	Dialed number string used by the MultiVoice gateway to complete the call. (Ascend Trap 24)
Service-Type	Requested type of service, the value of the Type of Service byte, for this call. (RFC Trap 6)

Attribute	Specifies
Ascend-H323-Destination-NAS-ID	NAS IP address used to route the call to the connecting MultiVoice gateway. (Ascend Trap 22)
Ascend-H323-Gatekeeper-IP	IP address of the MultiVoice gatekeeper used to route the call. The MultiVoice gateway is registered with this MultiVoice gatekeeper. (Ascend Trap 19)
Ascend-Global-Call-Id	IP address used by the MultiVoice gatekeeper to identify the connecting MultiVoice gateway for this call. (Ascend Trap 20)
Ascend-H323-Conference-ID	IP address used to identify the called destination. (Ascend Trap 21)
Ascend-H323-Presession-Time	Time from the moment the caller finishes dialing the destination telephone number until the moment the speech path is established to the called destination. (Ascend Trap 198)
Ascend-H323-Dialed-Time	Time the user spends dialing the destination telephone number. This value is zero for call originating from the LAN. (Ascend Trap 23)
Ascend-Session-Type	Audio codec used for processing the call. (Ascend Trap 18)

Stop records

A stop record is generated at the moment when MultiVoice begins to tear down the speech path or when an incoming call to a MultiVoice gateway fails to connect. A stop record can contain following information:

Attribute	Specifies
Acct-Session-Time	Time from the moment the speech path is established to the called destination until the moment MultiVoice begins to tear down the speech path. (RFC Trap 46)
Ascend-Connect-Progress	Number that represents the call connection state at the time the call was terminated. (Ascend Trap 195)
Ascend-Disconnect-Cause	Number that reports the H.323 call disconnection reason. (Ascend Trap 196)
Ascend-H323-Inter-Arrival-Jitter	Estimated interarrival jitter for voice packets received by a MultiVoice gateway. (Ascend Trap 25)
Ascend-Dropped-Octets	Number of voice frames (in bytes) dropped by a MultiVoice gateway during call processing. (Ascend Trap 26)
Ascend-Dropped-Packets	Number of voice packets dropped by a MultiVoice gateway during call processing. (Ascend Trap 26)
Acct-Input-Octets	Number of voice frames (in bytes) received by a MultiVoice gateway during this call. (RFC Trap 42)
Acct-Input-Packets	Number of voice packets received by a MultiVoice gateway during this call. (RFC Trap 47)

Attribute	Specifies
Acct-Output-Octets	Number of voice frames (in bytes) sent by a gateway during this call. (RFC Trap 43)
Acct-Output-Packets	Number of voice packets sent by a gateway during this call. (RFC Trap 48)

Call Progress records

A call progress record can be generated during a VoIP call when a change in resource occurs for a fax or transparent modem call. For fax calls, the record includes the modem speed and modulation. A progress message contains all the information included in a start record.

Storing voice announcements in the FAT-16 flash memory file system

This enhancement allows creation of multiple voice announcement directories on the flash memory file system and specifies a location for voice announcement files. After creating the directory on a flash card and moving voice announcement files into it, specify the pathname in the Voice-Ann-Dir setting.

By default, MultiVoice callers are notified of call progress by dual tone multifrequency (DTMF)-based tones. The tones report easily recognized call states such as ringback, busy signal, and so forth, as well as tones specific to MultiVoice, such as PIN prompt, which are not as easily recognized by callers. Previous MultiVoice releases introduced support for playback of custom voice announcements to callers to indicate call progress. For details about how voice announcements work, and for information about managing them in the MAX TNT, see the *MultiVoice for the MAX TNT Configuration Guide* at `http://www.lucent.com/ins/doc/library`.

Using the Voice-Ann-Dir parameter, you can create up to four directories on the external flash memory card for customized voice announcements for playback to report call progress or on command from the MultiVoice Access Manager (MVAM), Lucent Technologies Softswitch, or a third-party billing application.

For example, the following commands create a directory named `messages` and a subdirectory named `announce` on the flash card in slot 1:

```
admin> mkdir 1/messages
admin> mkdir 1/messages/announce
```

The following command loads a voice-announcement file named `busy.au` from a TFTP server at 10.10.10.10 to the `/current` directory on flash card 1 (flash card 1 is the default):

```
admin> load file network 10.10.10.10 busy.au
```

The following command moves the `busy.au` file to the new subdirectory on flash card 1:

```
admin> mv 1/current/busy.au 1/messages/announce/busy.au
```

The following commands inform the MultiVoice subsystem of the location of the voice announcement files:

```
admin> read voip { 0 0 }
VOIP/{ 0 0 } read
admin> set voice-ann-dir = /messages/announce
```

```
admin> write
VOIP/{ 0 0 } written
```

You can specify a pathname up to 40 characters long. When the system receives a request to play an announcement, it looks in the specified directory on the flash card in slot 1. If the card is not present or the voice announcement file is not found, the system looks for the specified directory on flash card 2.

MultiVoice: Compress a two-frame VoIP packet into three ATM cells

TAOS 9.0 reduces the number of backplane asynchronous transfer mode (ATM) cells used by MAX TNT units to transmit a packet containing two frames of G.711-encoded voice from four cells to three cells. This reduction allows the ATM backplane to more adequately support voice call volume.

This enhancement allows more efficient utilization of the available ATM backplane bandwidth by reducing the size of the packet bus and packet bus header, improving the packet transmission rate.

To fit a two-frame G.711-encoded RTP packet into three ATM cells across the MAX TNT unit backplane, TAOS reduces the size of the packet bus header from 36 or more octets to 16 octets by doing the following:

- Removing unused or derivable fields from the packet bus header.
- Creating a unique IP packet bus client for processing IP protocol packets, which include the packets generated for VoIP call processing.
- Creating unique input and output packet bus headers. The packet bus header uses parameters that are unique to incoming packets (those flowing from a slot card in to the shelf controller or another slot) and outgoing packets (those flowing from the shelf controller or a slot out another slot).

MultiVoice: Support of Full Rate GSM audio codec

This enhancement add support for the Full Rate Global System for Mobile Communication (GSM) audio codec as defined by ETSI Recommendation GSM 06.10, *GSM Full Rate Speech Transcoding*, (Feb. 1992), European Telecommunications Standards Institute (ETSI).

Overview of Full Rate GSM

Full Rate GSM is a voice encoder/decoder standard for cellular communications. It compresses the speech samples from 64-Kbps pulse code modulation (PCM) to 13.2 Kbps, requiring less network bandwidth than G.711 A-Law or U-Law. Full Rate GSM is the standard followed for European, Japanese, and Australian cellular communications systems, and is supported by certain Web telephone applications. Full Rate GSM uses a speech frame size of 160 samples (20 ms), and the encoder produces 33 bytes per frame. The decoder produces 160 samples (20 ms) of speech from the 33-byte encoder output.

This enhancement also supports Silence Detection and Comfort Noise Generation for Full Rate GSM, as defined by the following recommendations:

- ETSI Recommendation GSM 06.12, *Comfort Noise Generation*, (Feb. 1992), European Telecommunications Standards Institute

- ETSI Recommendation GSM 06.12, *Discontinuous Transmission* (Feb. 1992), European Telecommunications Standards Institute

A MAX TNT unit acting as a MultiVoice gateway reports Full Rate GSM during H.245 capability negotiation. Suppose both H.323 end points (such as a MultiVoice gateway and PC, or two MultiVoice gateways) choose Full Rate GSM as the preferred codec. Then, after opening the H.245 logical channel between them, both H.323 end points use Full Rate GSM for processing the VoIP call. Full Rate GSM is encoded as a standard audio capability.

User interface changes

To support the Full Rate GSM codec, this release adds `frgsm` as an option for the Packet-Audio-Mode parameter in the VoIP { x x } profile. Assigning this value configures the MultiVoice gateway to select the Full Rate GSM as the preferred audio codec for processing voice data for VoIP calls.

The following commands configure a MAX TNT unit configured as a MultiVoice gateway to use Full Rate GSM:

```
admin> read voip { 0 0 }
VOIP/{ 0 0 } read

admin> set packet-audio-mode = frgsm

admin> write
```

Changes to the Packet-Audio-Mode parameter are effective with the next VoIP call.

MultiVoice: Support for G.729-encoded voice announcement files

This enhancement adds support for using either G.729-encoded or G.711-encoded speech for voice announcement playout on MAX TNT unit acting as a MultiVoice gateway. This feature is configurable through the TAOS administration interface.

Audio encoded in G.729 format is eight times smaller than audio encoded in G.711. This feature allows customers to store and play a larger number of voice announcements.

Note: While G.711-encoded voice announcement files can be created with standard off-the-shelf software, a special tool is need to create G.729-encoded voice announcements that the MultiVoice gateway can recognize and play. This tool is available free to customers from the Lucent Technologies FTP download site. Contact your account representative for details.

This enhancement adds the Voice-Ann-Enc parameter to the VoIP { x x } profile.

Voice-Ann-Enc

Description: Specifies encoding of voice announcements played out by a MAX TNT unit acting as a MultiVoice gateway where voice announcements are used for reporting call progress to callers.

Usage: When a MultiVoice gateway uses voices announcements to report call progress to callers, selecting `g711-ulaw` as the value for the Voice-Ann-Enc parameter enables use of G.711-U-Law encoding for voice announcement playout. Selecting `g729` as the value for this parameter enables use of G.729 encoding for voice announcement playout.

Example: `set voice-ann-enc = g729`

Dependencies: The Voice-Ann-Enc parameter has the following dependencies:

- The MultiVoice gateway must be configured for using voice announcements to report call progress.
- Before a MultiVoice gateway is configured to use G.729 voice announcement encoding (`voice-ann-enc=g729`), voice announcement files must be converted to the G.729-compatible format. Lucent Technologies offers a tool, at no charge to MultiVoice Customers, that creates G.729-encoded voice announcement files.
- The MultiVoice gateway must be configured to use G.729 voice announcement encoding (`voice-ann-enc=g729`) when the Lucent Technologies prepaid billing message set is used for reporting call progress and for billing announcements.
- Changes to the Voice-Ann-Enc parameter are effective with the next VoIP call.

Location: Voip { 0 0 }, Voip { x x }

MultiVoice: Arbitrary announcement playback and tone collection

This enhancement expands the voice announcement playback capability of MultiVoice gateways for break-in announcements and message queuing in response to either caller-entered DTMF signals or time-out or time-delay intervals. This capability expands MultiVoice support for third-party billing and prepaid billing applications and call queuing services.

MultiVoice can playback multiple voice announcements, in response to an Information Request (IRQ) message sent by the MultiVoice Access Manager (MVAM) to a MultiVoice gateway.

By restructuring the message request/reporting fields in the nonStandardData byte of the Information Request (IRQ) messages exchanged between MultiVoice gateways and the server running MultiVoice Access Manager (MVAM), you can present callers with voice menus and prompts that respond to caller input using DTMF tone collection. Voice menus and prompts give customers a mechanism for providing automated attendant functions on their MultiVoice networks. The call services are activated by DTMF entries.

Requests to play specific messages to callers are initiated by MVAM or in response to caller-entered digits. Message initiation is tied to call progress or user-entered DTMF tones sent by the MultiVoice gateway to the access manager. Message selection by MVAM is controlled through the MultiVoice application programming interface (API).

When processing voice announcement playout requests from MVAM, the MultiVoice gateway

- Acknowledges receipt of the IRQ containing the playout request
- Acknowledges playout of the message
- When collecting caller-entered DTMF tones, if appropriate, plays messages in response to DTMF entries
- When collecting caller entered DTMF tones, if appropriate, plays messages after a pre-defined time-out or time-delay interval expires when no DTMF entries are collected
- Reports collected DTMF strings to the MVAM for further processing by third-party billing, prepaid billing, or other applications that use the MultiVoice API to perform call administration

When requesting voice announcement play out from the MultiVoice gateway, MVAM

- Acknowledges receipt of Information Request Response (IRR) message containing the voice announcement play-out results, including collected DTMF strings
- Report collected DTMF strings to any third-party billing, prepaid billing or other applications utilizing the MultiVoice API to perform call administration
- Sends the next play message, when appropriate, in response to results reported in an IRR message
- Sends requests to break in with new announcements, even when a previously requested announcement is still playing

Deactivating trunks used for VoIP calls

The trunk deactivation feature enables MAX TNT units acting as MultiVoice gateways to automatically deactivate trunks used for VoIP calls when a MultiVoice gateway becomes unavailable. This feature allows Gatekeepers in the MultiVoice network to route calls to other available MultiVoice gateways, to use network resources more efficiently and improve service quality for users.

Note: In this release, only T1 trunks that use ISDN PRI signaling and have been configured for VoIP can be deactivated system-wide by means of this feature.

Trunk deactivation prevents the PSTN switch from routing subsequent calls to the trunks configured for VoIP. Current calls remain active until those calls are terminated by the caller or PSTN. When trunk deactivation is enabled, trunks configured to accept VoIP calls are made unavailable to the PSTN under the following conditions:

- A MultiVoice gateway cannot register with either a primary or secondary Gatekeeper.
- A MultiVoice gateway's trunk connection with the PSTN is unavailable, so that the MultiVoice gateway is forced to unregister itself from its Gatekeepers.

Previously, when a MultiVoice gateway could not register with the primary and secondary Gatekeeper, the caller heard a fast busy signal because the PSTN switch continued to route calls to the trunks on that MultiVoice gateway. Deactivating the trunk changes the trunk state to inform the PSTN switch that those trunks are not available.

Previously, when a VoIP call could not connect because a trunk was not operating, the caller heard a fast busy signal, because the Gatekeeper continued to route calls to that MultiVoice gateway as long as it remained registered. Deactivating the trunk forces the MultiVoice gateway to unregister from all known Gatekeepers, which causes the Gatekeepers to reroute new calls to other MultiVoice gateways. When any one of the MultiVoice gateway's trunks comes back in service, that MultiVoice gateway starts registering itself with one of its known Gatekeepers. The Gatekeeper then begins to route calls to this MultiVoice gateway.

This enhancement adds the Trunk-Quiesce-Enable parameter to the VoIP profile.

Trunk-Quiesce-Enable

Description: Enables automatic trunk deactivation whenever a MAX TNT acting as a MultiVoice gateway is unable to register with either a primary or secondary MultiVoice Access Manager (MVAM), or forces a MultiVoice gateway to unregister whenever the trunk connection to the PSTN is unavailable.

Usage: Assigning the value `yes` to the Trunk-Quiesce-Enable parameter causes the MultiVoice gateway to make itself unavailable to accept calls whenever it becomes unregistered or loses the connection to the PSTN. Assigning the value `no`, the default, allows it to continue processing call requests when unregistered or when its PSTN connection goes down.

Example: `set trunk-quiesce-enable = yes`

Location: Voip { 0 0 }, Voip { x x }

MultiVoice: Support for CLID substitution and early-ringback

This enhancement provides support for

- ANI/CLID substitutions for calling certain destinations
- Enabling early-ringback for use with high-latency network configurations

ANI/CLID substitution

When MultiVoice gateways are connecting VoIP calls, they can transmit a calling line ID (CLID) generated by the MVAM software on the Gatekeeper instead of the PSTN-generated CLID collected on the trunk line. CLID substitution allows the MultiVoice network to provide the appropriate E.164 address for both the called and calling telephone numbers to the respective PSTN, and for use by external applications.

In certain configurations in which the MultiVoice gateway connecting the call reside in different area codes or countries, the CLID received from the PSTN must be changed to provide the appropriate calling number information to the local carrier, or to call management and billing applications.

When the MVAM receives the CLID from a MultiVoice gateway, it translates the CLID to the appropriate dial string, adding or removing country codes and area codes as appropriate for the respective locations of the callers. The Gatekeeper then reports the revised CLID to the MultiVoice gateways as part of the admission confirmed (ACF) message.

Enabling early ringback

For certain VoIP network configurations, such as satellite IP networks, wireless networks, or networks using channel-associated signaling (CAS) trunks, call setup times can be quite long. Callers might hang up before the call completes because they hear no call progress tones until RTP carries ringback from the far-end PSTN. Early ringback allows the MAX TNT to generate a ringback tone locally, as soon as the call is started on the far-end MultiVoice gateway.

Note: Early ringback is intended for use only on networks that experience long call setup times. Its use for other network configurations is not recommended, and might result in erroneous ring-to-busy and ring-to-failure announcements.

User interface changes

This enhancement adds the Early-Ringback-Enable parameter to the VoIP profile.

Early-Ringback-Enable

Description: The Early-Ringback-Enable parameter is used to enable local generation of a ringback tone when call startup begins on an egress MultiVoice gateway. When enabled, MultiVoice can alert a caller that call setup is in progress, while waiting on a connection to the PSTN. This is designed for high-latency networks, where response from the PSTN may be delayed.

Usage: When enabled (`early-ringback-enable=yes`), the MultiVoice gateway generates a ringback tone for the caller once call setup begins on the egress MultiVoice gateway. When disabled (`early-ringback-enable=no`), the default, MultiVoice waits for the PSTN to begin generating ringback tones before passing them to the caller.

Example: `set early-ringback-enable = yes`

Location: Voip { 0 0 }, Voip { x x }

MultiVoice: Support for T.38 and transparent fax/modem for IPDC

This enhancement enables support for a transparent data mode that enables users to run a modem on an SS7 VoIP channel using IPDC, regardless of the codec that is in use. Previously, support for this feature was available only for H.323-controlled PSTN calls. This enhancement includes support for dynamic echo canceller control on a per-call basis via IPDC.

T.38 Fax

T.38 fax is used to carry facsimile traffic over an IP link. Currently, T.38 is only employed over H.323 VoIP calls. New IPDC messages allow the Softswitch to request the MAX TNT to enter T.38 fax mode upon fax tone detection from the MultiVoice gateway.

Note: The Rt-Fax hash code is required to enable T.38 fax for IPDC.

Transparent Data

MultiVoice gateways detect fax/modem tones in both the TDM connection and RTP stream. When fax/modem tones are detected, echo cancellation and suppression are automatically disabled. When codecs other than the G.711 A-Law and U-Law are used, IPDC messages allow the Softswitch to request the MAX TNT to enable G.711 transparent data mode upon fax tone detection from the MultiVoice gateway.

Echo Canceller

Echo Cancellation Tag (0x74) is implemented on a per-call basis. Only values of 0 (off) and 32 ms are currently supported.

Details of IPDC message support

Changes to existing message tags

The existing message tag values shown in Table 19 are modified for the NTN message to support T.38 and transparent fax/modem detection.

Table 19. Modified NTN message tag values

Tag	Description	Values
0x33	Tone string	• f : Fax tone (CED, no phase reversal, or V.21 flags) • o : Modem tone (CED, phase reversal)
0x49	Tone type	• 0x06: Fax tone (CED, no phase reversal, or V.21 flags) • 0x07 : Modem tone (CED, phase reversal)

The existing message tag values shown in Table 20 are modified for the Request Packet Pass-Through Call (RCCP), Accept Packet Pass-Through Call (ACCP), Request Modify Packet Pass-Through Call (RMCP), and Accept Modify Packet Pass-Through Call (AMCP) messages to support T.38 and transparent fax/modem detection.

Table 20. Modified RCCP, ACCP, RMCP, and AMCP message tag values

Tag	Description	Values
0x70	Encoding type	• 0x60 :Transparent Data encoding • 0x61 : T.38 Fax over UDP

Note: Currently, Transparent Data is nothing more than G.711 RTP with some things turned off. However, this could be something different in the future, something similar to T.38 for fax.

New message tag values

The new message tag values shown in Table 21 are added to the RCCP, ACCP, RMCP, and AMCP messages to support T.38 and transparent fax/modem detection. These values are applied on an individual call basis.

Table 21. New RCCP, ACCP, RMCP, and AMCP message tag values

Tag	Description	Values
0x74	Echo Cancellation	• 0x00 : Echo canceller off (0 ms) • 0x01 : Echo canceller on (32 ms)
0x77	Constant Fax tone detection	Reports which fax tone support is enabled (either `rt-fax-enable=yes` or `g711-trans-data=yes`) and overrides this setting if appropriate.
0x78	Constant Modem tone detection	Reports whether modem tone support is enabled (either `g711-trans-data=yes` or `g711-trans-data=no`) and overrides this setting if appropriate.

New message tags

The NTN message is sent by the MultiVoice gateway to Softswitch when fax or modem tone detection is enabled and either tone is detected. The fax/modem tone detection can be enabled or disabled by either IPDC tags in RCCP and RMCP messages, or in the VoIP profile.

The new message tag value shown in Table 22 is added to support T.38 and transparent fax/modem detection. These values are applied on an individual call basis.

Table 22. New NTN message

Tag	Description	Values
0x00F0	Notify Tone	This NTN message from the MAX TNT notifies Softswitch of asynchronous fax/modem tone detections.

The tags shown in Table 23 can be included in a Notify Tone message.

Table 23. Notify Tone message tags

Tag	Parameter Description	Status
0x65	Source Post Type	Required
0x07	Source Module Number	Required
0x0D	Source Line Number	Required
0x15	Source Channeled Number	Required
0x40	Ascend Route ID	Optional
0x33	Tone String	Required
0x49	Tone Type	Required

Currently Tone String and Tone Type convey the same information.

User interface changes

This enhancement includes modifications to the `ss7nmi -m` command output to include modifications for the RMCP, AMCP, and NTN messages as illustrated by the following:

```
tnt15>ss7nmi -m
IPDC message processing statistics:
   Message code      Received       Sent
   RCR   (0x0011):          1          0
   ACR   (0x0012):          0          1
   RCCP  (0x0013):          1          0
   ACCP  (0x0014):          0          1
   RMCP  (0x0015):          1          0
   AMCP  (0x0016):          0          1
   RMS   (0x0041):          1          0
   NMS   (0x0042):          0         17
   RLS   (0x0043):          1          0
   NLS   (0x0044):          0          1
   NSUP  (0x0081):          0          1
   ASUP  (0x0082):          1          0
```

```
   NTN   (0x00f0):          0          1

Data collection was started: [04/26/2000 15:40:47]
```

MultiVoice: Support for transparent fax/modem over VoIP

This feature enables a MultiVoice gateway to process fax/modem traffic over a VoIP channel, regardless of which audio codec is currently in use.

Feature description

This feature enables a MultiVoice gateway to detect a fax/modem transmission on a VoIP channel, and enable fallback to the G.711 audio codec to allow transparent processing of fax/modem transmission. Detection of fax/modem is based on an algorithm that listens for a fax/modem Answer tone, generated by an answering fax/modem. The Answer tone is significantly different for high-speed modems and fax terminals. The difference in Answer tones allows a MultiVoice gateway to discriminate between the two types of equipment. Typically both real-time fax and transparent data can be enabled simultaneously.

To work, this feature must be enabled on both MultiVoice gateways connecting the fax/modem call. Both MultiVoice gateways must agree to transparent mode before the call bandwidth is increased to G.711 bandwidth of 64 Kbps.

Using transparent modem with real-time fax

If the MAX TNT has been licensed for real-time fax, users can run either a high-speed modem with speeds greater than 2400 bps or a fax terminal in the VoIP channel. This capability provides a fallback for real-time fax transmissions. Both fax terminals and high-speed modems transmit a single tone when they answer a call, but each type of equipment uses a different tone. The MAX TNT detects the type of equipment in use on the basis of its answer tone. When it detects the equipment answering the call, the MAX TNT sends H.245 request-mode messages to request a switchover from the current audio codec to either G.711 with no echo canceler (for transparent modem) or T.38 data mode (for real-time fax).

Transparent data is encoded as an audio-mode type, either G.711 U-Law (64Kbps) or G.711 A-Law (64Kbps). Real-time fax (if supported) is encoded as data-mode type T.38 fax.

Note: Transparent data mode introduces an H.245 request-mode message that is not backward compatible with the real-time fax feature provided by releases earlier than TAOS 8.0. For this reason, Lucent Technologies recommends upgrading to TAOS 9.0 all systems that must interoperate with a MultiVoice gateway using transparent mode.

Limitation for low-speed modems

Real-time fax cannot be used concurrently with low-speed modems (2400bps or less) because these modems use the same answer tone as fax terminals. If a low-speed modem is used on a VoIP channel that is enabled for real-time fax, the MultiVoice gateway detects a fax answer tone and requests T.38 encoding. The ingress MultiVoice gateway (typically the gateway on which the modem call originated) can accept the T.38 encoding request or reject the request, which causes the egress MultiVoice gateway to terminate the call.

User interface changes

This release adds the G711-Transparent-Data parameter to the VoIP { x x } profile, as illustrated:

```
[in VOIP/{ 0 0 }]
voip-index* = { 0 0 }
gatekeeper-ip = 135.92.52.138
gk-mlg-control = no
vpn-mode = no
single-dial-enable = no
packet-audio-mode = g729
frames-per-packet = 4
...
g711-transparent-data = no
...
```

G711-Transparent-Data

Description: The G711-Transparent-Data parameter is used to enable or disable transparent transmission of fax or modem signals across VoIP channels. When enabled, if a MultiVoice gateway detects a fax or modem Answer tone in a VoIP channel, the unit transparently requests end-to-end G.711 encoding and bandwidth for the call, in a process similar to that used by real-time fax. The echo cancelers are disabled when the MAX TNT enters this mode, thus providing transparent G.711 encoding. The data is encoded transparently as an audio-mode type, either G.711 U-Law (64Kbps) or G.711 A-Law (64Kbps).

Usage: The G711-Transparent-Data parameter accepts the following values:

Value	Description
yes	When this value is used, a MultiVoice gateway transparently requests end-to-end G.711 encoding and bandwidth for the call upon detection of a fax or modem Answer tone in a VoIP channel.
no	When this value is used, the default, a MultiVoice gateway continues with VoIP call processing, even when a fax or modem Answer tone is detected.

Example: The following commands enable the transparent modem feature on VoIP channels:

```
admin> read voip { 0 0 }
VOIP/{ 0 0 } read
admin> set g711-transparent-data = yes
admin> write
VOIP/{ 0 0 } written
```

The following commands enable both real-time fax and the transparent modem feature for high-speed modems:

```
admin> read voip { 0 0 }
VOIP/{ 0 0 } read
admin> set g711-transparent-data = yes
admin> list rt-fax-options
admin> set rt-fax-enable = yes
```

```
admin> write
VOIP/{ 0 0 } written
```

Dependencies: The G711-Transparent-Data parameter is `N/A` when either G.711 U-Law or G.711 A-Law encoding is selected for the Packet-Audio-Mode parameter (such as: `packet-audio-mode=g711-alaw`).

Location: Voip { 0 0 }, Voip { x x }

MultiVoice: Support for modem and VoIP cohabitation

TAOS 9.0 introduces the *cohabitation* feature on the MultiDSP slot card for MAX TNT units. Cohabitation refers to the ability to run two applications on a single Digital Signal Processor (DSP) in the following combinations

- One VoIP session using either the G.729 or G.711 audio codec, and one modem session
- Two VoIP sessions using either the G.729 or G.711 audio codec
- Two modem sessions

The G.723.1, G.728, RT-24, and Full Rate GSM audio codecs are also supported under cohabitation, but require two DSP channels.

Cohabitation enables a MAX TNT unit to support multiapplication processing on the same platform for a combination of voice and data calls. During the call setup process, the StrongARM processor allocates DSPs to either voice or data calls depending upon

- Call type
- Requested audio codec
- Available DSP channels

Cohabitation is restricted to performing VoIP calls plus one other data call type, such as modems.

DSP allocation

As call requests are processed by the TAOS unit, the StrongARM processor on the MultiDSP card checks each incoming call to determine an application type and subtype. This information is used to determine how DSPs are allocated for that call. Modem and VoIP calls, regardless of the audio codec requested, have the same application type. The application subtype is different for the complex audio codecs.

Modem, G.729, and G.711 calls all belong to the modem application type and no application subtype. For cohabitation processing, when this application type and subtype are detected, only one DSP channel is allocated for the call. The twin channel on the DSP is assigned the no application subtype, and is considered available for processing other calls.

Calls using G.723, G.728, RT-24, and full rate GSM codecs have the application subtype VoIP. When that application subtype is detected, a whole DSP is allocated for the call.

Audio codec selection

Audio codec selection is determined during H.245 terminal capabilities between the two MAX TNT units that connect a call. When the call request is received at the MultiDSP slot card, the call is brought up initially as a modem application type. If the call request asks for

one of the complex audio codecs, the complex codec is loaded on a DSP where both channels are available.

Jitter buffer and packet redundancy for real-time fax operations

TAOS 9.0 adds a packet redundancy scheme and jitter buffer to improve performance of MultiVoice real-time fax over unmanaged networks such as the public Internet.

This enhancement allows the MultiVoice gateway to process several hundred milliseconds of packet jitter and allows the optional transmission of redundant packet data for fax calls across networks experiencing instances of packet loss and packet jitter.

To enable this feature, you must enable real-time fax support on the MultiVoice gateway. You can verify if the real-time fax feature is supported by checking the Base profile for the `rt-fax-enabled=yes` entry.

Packet redundancy

Redundant packet data is defined as the last *n* packets transmitted appended to the current packet. The value of *n* is set through the command line interface (CLI) with the Packet-Redundancy parameter.

Assigning the Packet-Redundancy parameter a value (such as `packet-redundancy = 4`), causes the MAX TNT to append that number of previously sent packets onto the current packet. On networks experiencing measurable packet loss, this enhancement improves the reliability of the fax transmission.

Depending upon the amount of measurable packet loss for a network, set the redundancy parameter as follows:

Network condition	Recommended value(s)
Packet loss occurs in frequent bursts.	1 - 5
Occasional packet loss (less than one percent)	0 (default)
Occasional packet loss (greater than one percent)	1 - 2

The additional bandwidth required for each fax call increases proportionally to the level of redundancy, adding 50 bytes of packet data per increment.

This enhancement uses a slip buffer to

- Allow MultiVoice Real-time fax to tolerate packet jitter
- Keep the modem fed with data, preventing modem underrun

Fixed-size packet format

The packet redundancy scheme uses a fixed-size packet format, consisting of a 49-byte payload, a prefixed sequence number, and a length field that precedes the payload data. When packet redundancy is enabled, *n*-length payload pairs are added at the end of the packet; *n* is the value of the Packet-Redundancy parameter.

Previously, the MAX TNT unit sent variable-length packets that were guaranteed to be zero terminated, allowing Class 1 modems to underrun gracefully.

TAOS 9.0 adds the Packet-Redundancy parameter to the RT-Fax-Options subprofile of the VoIP profile.

Packet Redundancy

Description: The Packet-Redundancy parameter causes a MAX TNT to append the designated number of previously sent fax packets onto the current packet. On networks experiencing measurable packet loss, this improves the reliability of the fax transmission.

Usage: This parameter accepts values from 0 through 5, directing MultiVoice to append the designated number of previously transmitted fax packets to the current packet, as follows:

Parameter value	Specifies
0	No change from the default packet behavior.
1	Append and send the previous fax packet with the current fax packet.
2	Append and send the two previous fax packets with the current fax packet.
3	Append and send the three previous fax packets with the current fax packet.
4	Append and send the four previous fax packets with the current fax packet.
5	Append and send the five previous fax packets with the current fax packet.

The following example illustrates how to change the default value of the `packet-redundancy` parameter.

```
admin> read voip { 0 0 }
VOIP/{ 0 0 } read

admin> set packet-redundancy=4

admin> write
VOIP/{ 0 0 } written
```

Dependencies: The following dependencies apply to this parameter:

- Once saved, packet redundancy is enabled with the next VoIP call.
- This value is set to `N/A` when `fixed-packets=no`.

Location: Voip{x x}>Rt-Fax-Options

MultiVoice: Support for real-time fax backward compatibility

With TAOS 9.0, you can disable the jitter buffer and packet redundancy scheme for real-time fax calls introduced in previous software releases. With packet redundancy disabled, a MultiVoice gateway running a software version earlier than TAOS 9.0 can process real-time fax calls to and from a MultiVoice gateway running the TAOS 9.0 software.

Feature definition

The packet sequence numbering introduced in previous TAOS releases for real-time fax required a format change to enable support for high-speed data packets. Without the high-speed packets, such as when a fax call was initiated from a MultiVoice gateway running an earlier version of TAOS, the MultiVoice gateway interpreted image data as sequence data. In addition, the smaller packets relied on the slip buffer to keep the modems with data or it dropped the carrier.

TAOS 9.0 introduces the Fixed-Packets parameter in the Rt-Fax-Options subprofile of the VoIP profile. Setting this parameter to `yes` (the default) disables the use of redundant packets and the slip buffer, and enables the MAX TNT to enable the pre-9.0 fax packet scheme for real-time fax processing. Fax calls are processed with variable-length packets that are zero-terminated, allowing Class 1 modems to underrun gracefully.

Setting this parameter to `no` enables jitter buffering and packet redundancy for real-time fax processing.

Example: `set fixed-packets=no`

Dependencies: The following dependencies apply to this parameter:

- Once saved, the selected packeting scheme is enabled with the next fax call.
- When this value is set to `yes`, then `packet-redundancy=n/a`.

Location: Voip{x x}>Rt-Fax-Options

MultiVoice: Real-time fax maximum data transmission rate limit

This enhancement makes the maximum data transmission rate allowed for a T.38 fax session configurable on a MultiVoice gateway. This provides customers with a means to regulate the bandwidth used for fax sessions on their networks.

Feature definition

This enhancement adds the capability to modify the rate negotiation between the originating and destination fax terminals through the MultiVoice gateway administration interface. This improves the reliability of the fax transmission by selecting lower fax transmission rates, resulted in fewer lost or repeated fax packets, and requiring less bandwidth for fax transmissions.

TAOS modifies the fax transmission rate by modifying the content of the Digital Identification Signal (DIS) frame transmitted from the destination fax, using the Max Rate parameter in the RT Fax Options profile. Upon receipt of that DIS frame, the originating fax uses the data transmission rate specified in the Max Rate parameter (or slower), and a supported modulation type. The content of the DIS frame is defined in the ITU Telecommunication sector standard (ITU-T) T.30, *Procedures for document facsimile transmission in general switched telephone networks*.

Changing the Max Rate parameter modifies the high-speed data transmission rate reported by the destination fax, and masks certain modulation types associated with higher fax transmission speeds. For example, once the data rate is set for 9600 bps, V.17 and V.33 are disallowed even though V.17 supports 9600 and 7200 bps. This is necessary because the DIS frame can specify only the supported modulation types for the highest selected transmission

speeds on the destination fax, and because the calling fax terminal requires “training” to match the supported modulation. The value assigned to the Max Rate parameter on the egress MultiVoice gateway sets the maximum fax transmission rate for the call.

User interface changes

This enhancement adds the Max-Rate parameter in the Voip { X X }>RT-Fax-Options profile. as illustrated by the following example:

Max-Rate

Description: The Max-Rate parameter allows MutliVoice to modify the rate negotiation between the originating and destination fax terminals. This improves the reliability of the fax transmission by reducing the number of lost or repeated packets that occurs during high rate transmissions, and reduces the required bandwidth for fax transmissions.

Usage: Values assigned to this parameter cause MultiVoice to do the following:

Parameter value	Specifies
14400	Default. Mask the fax capabilities in the DIS frame that support fax data transmission at rates higher than 14,400 bps.
9600	Mask the fax capabilities in the DIS frame that support fax data transmission at rates higher than 9,600 bps.
4800	Mask the fax capabilities in the DIS frame that support fax data transmission at rates higher than 4,800 bps.
2400	Mask the fax capabilities in the DIS frame that support fax data transmission at rates higher than 2,400 bps.

Example: The following example illustrates how to set the fax data transmission rates:

```
admin> read voip { 0 0 }
VOIP/{ 0 0 } read

admin> list rt-fax-options
[in VOIP/{ 0 0 }:rt-fax-options]

admin> set max-rate=9600

admin> write
VOIP/{ 0 0 } written
```

Dependencies: This parameter has the following dependencies:

- This parameter is N/A when `rt-fax-enable=no`.
- Changes made to this parameter are enabled for the next VoIP call.

Location: Voip{X X}->Rt-Fax-Options

MultiVoice: Trunk prefixing

Trunk prefixing enables a MAX TNT gateway to identify and assign an egress trunk group to the destination telephone number. The trunk group prefix is used to select the egress trunk to connect the call after it has been received by the egress MultiVoice gateway or call signaling entity.

When trunk prefixing is enabled, the MAX TNT unit can obtain an egress trunk-group number from either of the following:

- The `Trunk-Group` parameter in the T1 Line profile associated with the inbound trunk on the ingress MultiVoice gateway
- The ACF message from the MultiVoice Access Manager (MVAM)

Once assigned, the trunk group number is prepended to the destination telephone number. The trunk group/dial string combination is sent as the Q.931 Called Party Number IE in the H.225/Q.931 SETUP message to the egress MultiVoice gateway. The destination address of the SETUP-UUIE is currently not encoded.

To support trunk prefixing, TAOS 9.0 adds the Trunk-Prefix-Enable parameter to the VoIP profile.

Trunk-Prefix-Enable

Description: Assigns trunk groups for connecting VoIP calls to the called end point on an egress MultiVoice gateway.

Usage: When set to `Yes`, the Trunk-Prefix Enable parameter causes an egress MultiVoice gateway to route outbound calls to the PSTN using a preselected trunk group, assigned by either the ingress MultiVoice gateway or the MAVM. When set to `No`, the default, the egress MultiVoice gateway selects trunk groups for outbound calls.

Example: `set trunk-prefix-enable = yes`

Dependencies: This parameter has the following dependencies:

- Trunk groups must be enabled on the egress MultiVoice gateway (`use-trunk-groups = no`).
- The size of the trunk groups must be defined (`num-digits-trunk-groups = 1`) on all egress MultiVoice gateways.
- Trunk group numbers must be assigned in both the T1 trunk and line profiles for egress T1 trunks.

MultiVoice: DTMF tone processing over R2 signaling

TAOS 9.0 enables MAX TNT units to process dual tone multifrequency (DTMF) tones over R2 signaling trunks for either country-specific R2 signaling (MFC-R2) or DTMF signaling over trunks supporting standard R2 signaling.

DTMF R2 signaling is generated by smaller European network switches and PBXs. TAOS 9.0 implements DTMF tone processing using the R2 signaling standard defined by the International Telecommunications Union Telecommunication Standardization sector standard (ITU-T) Q.400, *Specifications of Signaling System R2 Definition and Function of Signals—Forward Line Signals*.

A channelized E1 card uses one of the following channelized associated signaling (CAS) types:

- R1
- R2 or any R2 variant
- DTMF-R2

Note: Only one signaling type can be used on a MAX TNT unit channelized E1 slot card.

To support DTMF-R2 detection, a MAX TNT unit requires the following:

- Connection to E1 trunks must be attached to a switch that supports the ITU-T R2 signaling standard.
- The switch must generate and/or relay the high-frequency/low-frequency tone combinations generated by normal touchtone dialing to the MultiVoice gateway.
- E1/R2 signaling must be enabled on the MAX TNT unit. To verify that this signaling is enables, check the Base profile for the `r2-signaling-enabled=yes` parameter.

Detection of DTMF R2 signals is enabled from the E1 line profile.

DTMF tone detection

When processing tones for DTMF R2 signaling, the MAX TNT unit

- Upon detection of an inbound call, first allocates a digital signal processor (DSP) for detecting DTMF tones, then captures DTMF digits as they are received from the switch.
- Upon receipt of an outbound call (from the packet network), first allocates a DSP for generating DTMF tones, then sends the first DTMF tone for 70ms, followed by 70ms of silence. The tone/silence sequence is repeated until all digits are sent to the telephone switch.

User interface changes

To support DTMF-R2 signaling on channelized E1 cards, TAOS 9.0 adds a new setting to the Signaling-Mode parameter and two new parameters to the E1 profile.

Parameter	Description
Signaling-Mode	Setting the value of the Signaling-Mode parameter in the Line-Interface subprofile of the E1 profile to the new `dtmf-r2-signaling` value enables the MAX TNT unit to recognize and respond to the DTMF R2 signal set during voice and data calls. DTMF R2 detection begins with the next VoIP call.

Parameter	Description
	When `signaling-mode=dtmf-r2-signaling`: • Collect-Incoming-Digits must be enabled (`collect-incoming-digits=yes`). • Assigning a lower value (such as `600` to `3000`) to the T1-Inter-Digit-Timeout parameter improves call setup times. Assigning a higher value (such as `3001` to `6000`) improves DTMF detection. • DTMF R2 detection is supported only when R2 signal processing is enabled for a MAX TNT unit. The Base profile must contain the following setting: `r2-signaling-enabled=yes`
Collect-Incoming-Digits	Decoding of DTMF tones for incoming calls. `Yes` enables decoding. Default is `no`.
T1-Inter-Digit-Timeout	Amount of time MAX TNT unit allows an incoming call to wait after receiving the last DTMF digit. Range is 100 to 6000 ms. Default is `3000` ms.

The following example shows how to enable DTMF-R2 signaling on a MAX TNT E1 line slot card.

```
admin> read e1 { 1 1 7 }
E1/{ 1 1 7 } read
admin> set signaling-mode=dtmf-r2-signaling
admin> set collect-incoming-digits=no
admin> set t1-inter-digit-timeout=3000
admin> write
E1/{ 1 1 7 } written
```

Support for A-Law companding on DTMF DSP code

This enhancement adds support for A-Law companding to the DSP that decodes DTMF on the slot T1 card. The companding mode, which defaults to U-law, can be changed automatically when A-Law companding is required.

This enhancement allows a MAX TNT unit to automatically enable A-Law companding upon detection of the DTMF by the decode Digital Signal Processor (DSP) on a T1 card. This process is accomplished during call setup on the T1 card. As the call comes up on the T1, the DSP checks for a companding mode message that specifies A-Law companding. If no message is sent, the DSP defaults to U-Law. This feature allows existing software to use the new load without the need to change anything to add the new message.

MultiVoice: Support for E1 R2 variable-length DNIS without EOP

TAOS 9.0 supports the collection of variable-length dial strings without using end-of-pulse (EOP) signaling. To implement this feature, TAOS 9.0 uses a time-out followed by pulse signals, as specified by ITU-T Recommendation Q.442, *Specifications of Signalling System R2 interregister Signalling, Pulse Transmission of Backward Signals A-3, A-4, A-6 or A-15* (1993), International Telecommunications Union.

In certain areas outside the continental United States where E1 MFC-R2 signaling is used for switched network operations, the length of E.164 addresses vary. End-of-pulse (EOP) detection is inefficient because the network is sometimes unable to complete the call as a result of less-than-optimal network conditions. This enhancement allows a MultiVoice gateway to delay processing of a dialed number string, even after receiving the last digit, to allow the resources on the switched network additional time to become available, before continuing with call processing.

To support this feature, TAOS 9.0 adds a new setting for the Number-Complete and Inter-Digit-Time-Out parameters in the E1 profile.

Number-Complete

Description: The Number-Complete parameter sets the condition the MultiVoice gateway uses to determine the length of the dial string. For E1 MFC-R2, the MultiVoice gateway continues to collect digits until the on/off pulsing using to transmits the dial string is complete.

Usage: The Number-Complete parameter now accepts the following value:

Parameter value	Specifies
`time-out`	Sets the MultiVoice gateway to reset the network idle timer after the initial digit is received then wait for silence. Once silence is detected, waits the interval specified by the Inter-Digit-Time-Out parameter for next digit. The MultiVoice gateway continues to collect digits, while waiting for the network idle timer to expire before continuing with call processing.

Example: `set number-complete=time-out`

Dependencies: E1 MFC-R2 signaling is country specific. The Signaling-Mode parameter, and the Country parameter in the System profile, must be set for the country-appropriate signaling in order for the MultiVoice gateway to properly detect dialed digits.

Location: E1 { x x } > Line-Parameters

Inter-Digit-Time-Out

Description: The Inter-Digit-Time-Out parameter controls how long a MultiVoice gateway will wait after receiving the last digit of a dial string before declaring DNIS/ANI collection complete. When using inband signaling (T1, MF R2), a MAX TNT unit waits until this interval has elapsed to ensure it has received all audible tones used to transmit DNIS/ANI across the trunk.

Usage: The Inter-Digit-Time-Out parameter accepts values between 100 and 6000 ms. This parameter defaults to 3000 ms. (3 seconds). To support E1 MRC-R2 signaling, the Inter-Digit-Time-Out parameter now accepts values between 200 and 6000 ms.

Example: The following example illustrates how to configure the interdigit timer on a Multi-Voice gateway to wait 1 second (1000 ms) in between dialed digits before continuing with call processing.

```
admin> read e1 { 1 1 1 }
E1/{ 1 1 1 } read
```

```
admin> list line
[in E1/{ shelf-1 slot-1 1 }:line-interface]

enabled = yes
frame-type = esf
encoding = b8zs
signaling-mode = inband
.........
ss7-continuity = { loopback single-tone-2010 }

admin> set inter-digit-time-out=1000

admin> write
```

Dependencies: E1 MFC-R2 signaling is country specific. The Signaling-Mode parameter, and the Country parameter in the System profile, must be set for the country-appropriate signaling in order for the MultiVoice gateway to properly detect dialed digits.

Location: E1 { x x } > Line-Parameters

MultiVoice: Support for Feature Group D (FGD)

This enhancement implements a subset of the Telecordia requirements for Feature Group D (FGD) signaling for Voice over IP (VoIP) processing. This implementation supports passing the automatic number identification II (ANI II) information, calling-party number, and called-party number as MFR1 tones on inc-wink signaled trunks. It also enables a MAX TNT functioning as a MultiVoice gateway to manage interworking between access tandem (AT) carriers and traditional toll service carriers for VoIP calls.

To enable the FGD signaling feature, you must have a special FGD software license, available from Lucent Technologies, loaded onto your system.

FGD signaling

This implementation provides basic support for trunk-side access with equal access (EA) dialing capability, presubscription, and enhanced signaling options for automatic number identification as specified by Requirement GR-690-CORE, *Exchange Access Interconnection FAS 20-24-0000* (Oct. 1995), Telecordia Systems (formerly Bellcore).

Feature Group D (FGD) access service with equal access multifrequency signaling is characterized by two-stage outpulsing when connection is made through the access tandem (AT) switch. The first stage provides information to the access tandem switch for selection of a carrier and the route to take to that carrier. The second stage provides the carrier with both the calling-party number (and, optionally, automatic number identification) and the called-party number (address or destination number). Overlap outpulsing is used to transmit this information by means of multifrequency signaling.

Starting with this release, passthrough of equal access signaling can be enabled for T1 inband trunks from the TAOS T1 Line-Interface profile. When FGD signaling is enabled, MultiVoice gateways can recognize and process a single-dialed access carrier destination (for example, 1,2025551212 or 1,1010220,2025551212). To support access carrier billing, a MultiVoice gateway passes the calling-party number, automatic number identification (ANI) information digits, and called-party number. The ANI information digits are a two-digit code that classifies the calling-party number by tariff type (for example, coin, 800 service, or POTS).

This enhancement also allows MultiVoice to manage interworking when connecting VoIP calls between access tandem networks and traditional toll service networks. A MultiVoice gateway that handles outgoing calls can be configured to receive the calling-party number, ANI information digits, and called-party number from an access tandem switch and connect that call to a switched telephone network that supports Feature Group C (FGC) switching. FGC, which is traditional toll service, includes automatic number identification of the calling party, callback, and disconnection supervision. FGC service predates the breakup of the Bell System.

New Signaling-Mode parameter and values

To support the FGD signaling feature, this release adds a new read-only parameter, Fgd-Signaling-Enabled, to the Base profile, and four new values to the Signaling-Mode parameter in the Line-Interface subprofile of the T1 profile.

Fgd-Signaling-Enabled

Description: Indicates whether Feature Group D (FGD) signaling is enabled on a MAX TNT functioning as a MultiVoice gateway.

Usage: The FGD software licence must be loaded onto a MAX TNT unit for the FGD feature to be enabled. During manufacturing or software upgrade of MultiVoice gateways, the installation binaries used to install TAOS on the MAX TNT request whether Feature Group D support is to be enabled. To disable FGD support, you unload the FGD software license from the MAX TNT.

The Fgd-Signaling-Enabled parameter is read only. Yes indicates that FGD signaling is enabled. No indicates that FGD signaling is disabled.

Example: `fgd-signaling-enabled = yes`

Location: Base

New Signaling-Mode values for FGD

This release adds the following four new values to Signaling-Mode to help implement FGD signaling support on a MAX TNT acting as a MultiVoice gateway. The FGD software licence must be loaded onto a MAX TNT unit for the FGD feature to be enabled.

Parameter	Specifies	
Signaling-Mode	Type of call signal received from the incoming (ingress) switched telephone network, and the type of call signals passed to the outgoing (egress) switched telephone network. Valid values include the following:	
	`fgd-in-fgd-out-inband`	MultiVoice gateway can receive call signaling data in FGD format, and connect VoIP calls to the egress switched telephone network sending call signaling data in FGD format.
	`fgd-in-fgc-out-inband`	MultiVoice gateway can receive call signaling data in FGD format, and connect VoIP calls to the egress switched telephone network sending call signaling data in FGC (traditional toll service) format.

Parameter	Specifies	
	`fgc-in-fgc-out-inband`	MultiVoice gateway can receive call signaling data in FGC format, and connect VoIP calls to the egress switched telephone network sending call signaling data in FGC (traditional toll service) format.
	`fgc-in-fgd-out-inband`	MultiVoice gateway can receive call signaling data in FGC format, and connect VoIP calls to the egress switched telephone network sending call signaling data in FGD format.

The following example illustrates how to enable a MultiVoice gateway to receive call signaling data in FGD format, and send call signaling data to the egress switched telephone network in FGC format signaling using the Signaling-Mode parameter.

```
admin> read t1 { 1 1 1 }
T1/{ 1 1 1 } read

admin> list line
[in T1/{ shelf-1 slot-1 1 }:line-interface]

enabled = yes
frame-type = esf
encoding = b8zs
signaling-mode = inband
.........
ss7-continuity = { loopback single-tone-2010 }

admin> set signaling-mode=fgd-in-fgc-out-inband

admin> write
T1/{ 1 1 1 } written
```

Changes made to the Signaling-Mode parameter take effect with the next VoIP call.

FGD signaling timing

To ensure wide interoperability with available access tandem switches, MultiVoice uses the middle-range FGD signaling timing, which behaves as follows:

- Waits up to 210 ms for the first wink signal from the access carrier. Requirement GR-690-CORE specifies a range of 140 ms to 290 ms.
- Waits up to 5 seconds to receive the first digit. After sending the first wink signal on receipt of an off-hook (busy) signal, the MultiVoice gateways waits for 5 seconds before reporting a timeout error if the first digit signal is not received.
- Waits up to 4 seconds for a wink signal from the access carrier. Requirement GR-690-CORE specifies that access tandem switches must wait for up to 4 seconds for this signal.

Debugging FGD signaling

You can collect debugging information for FGD inband signal processing on a MAX TNT unit. For example, to turn on FGD signaling debugging on a T1 slot card in slot 1, enter the following commands:

```
admin> open 1 1

admin> fgdtoggle
```

```
admin> debug on
```

The debug command displays information similar to the following for FGD signal processing:

```
-------------------------------------------------------------------
0 Connections, 0 Sessions    | TNTFGD Status
                             | Serial number: 9340872 Version: 9.0a0e0
                             |
                             |  Rx Pkt:    1057139
                             |  Tx Pkt:     163995
                             |     Col:       2244
                             |
                             |
                             | 02/29/2000 17:50:27  Up: 7 days, 00:17:30
                               | "T3 Tru+ 1/03/00 LA la la la la la la la
                             | "T3 Fiv+ 1/03/01 LA F------- ------- -------
                             | "T3 CDX" 1/03/02 LA F-------- -------- --------
                             |                     ^
                             |                     |
                             |                     +-- display (F) for FGD line
                             |
                             |
-------------------------------------------------------------------------------
```

MultiVoice: Support for multiple logical gateways

This enhancement implements dynamic call control for H.323 VoIP calls on MultiVoice networks, and provides support for partitioning a single MultiVoice gateway into multiple logical gateways. Using this method of call control enables the H.323 Gatekeeper to perform call-specific administration on a call-by-call basis.

This feature allows call-specific administration of H.323 VoIP calls for the following call control functions on the same physical MultiVoice gateway:

- Personal identification number (PIN) prompting
- Single-stage dialing
- Two-stage dialing
- Voice announcement playback
- Configurable call timers for pre-paid and credit card billing systems

The MultiVoice Access Manager (MVAM) analyzes call performance data (trunk group, DS0 status and call activity) that it receives when a gateway performs periodic keep-alive registration. When MVAM responds to subsequent call requests from each gateway, the administration conformation (ACF) message includes any changes defined for the aforementioned call administration parameters. The gateway applies the parameter changes received from MVAM to the current call request. This information is stored as part of the nonstandard data included in registration, admission, and status (RAS) messages exchanged by the gateway and gatekeeper for each call.

Caution: Dynamic call control and multiple logical gateways are only supported in MultiVoice networks running TAOS 8.0-103 or later on the gateways and MVAM 3.0 on the gatekeepers. These features are not supported in MultiVoice networks where gatekeepers are running MVAM 2.x.

Previously, all H.323 call management features were configured globally on each MultiVoice gateway, using the values assigned in the VoIP Options profile. Now, using status information reported by MultiVoice gateways, a gatekeeper running MultiVoice Access Manager 3.0 can send instructions to the ingress gateway that override global call management settings. The decision to override the global call management settings may be based upon reported ingress trunk or DS0 groups, Caller ID, time-of-day, gateway, and so on.

The rules used to apply overrides to H.323 call-management parameters are configured on MVAM. These parameter changes are useful when partitioning MultiVoice gateways into logical gateways. *Logical gateways*, defined on MVAM, treat selected trunk groups on a MultiVoice gateway as if they were a unique VoIP gateway. Initially, MultiVoice gateways must have T1, T3, and PRI trunks to support logical gateways.

Note: While BRI lines can still be used for VoIP, the multiple logical gateway features are not supported on MultiVoice gateways using BRI.

A MultiVoice gateway cannot detect its logical gateways—only MVAM can. However, a gateway must be configured to apply instructions received from MVAM when processing the current call.

H.323 call-specific administration

H.323 call-specific administration enables the MultiVoice Access Manager (MVAM) to override defaults for PIN authentication, dialing mode, and voice announcement playback.

MVAM enables call-specific administration based upon the reported DNIS, ANI, trunk group, DS0 information, or any combination of that data, which are all reported in the first ARQ from the gateway.

Dynamic PIN authentication

When the multiple logical gateway feature is enabled on a MultiVoice gateway, any incoming call request immediately sends an ARQ to MVAM, which includes

- DNIS, when available
- ANI, when available
- Trunk group and DS0 status changes

If the ARQ includes all the information necessary to route the call, MVAM sends an ACF message to the gateway. The gateway then processes the call as if the following VoIP parameters were set to these values:

```
vpn-mode=yes
single-dial-enable=yes
```

If MVAM, or a third-party billing application used with MultiVoice, requires PIN authentication for the call, an admission reject (ARJ) message is issued, which directs the gateway to set `vpn-mode=no` for this call. The gateway then resumes call handling as if the call had just arrived from the PSTN, but prompts for authentication (as if `vpn-mode=no`) before continuing with call processing.

Dynamic single-stage and two-stage dialing

When the multiple logical gateway feature is enabled on a MultiVoice gateway, any incoming call request immediately sends an ARQ to MVAM, which includes:

- DNIS, when available
- ANI, when available
- Trunk group and DS0 status changes

If the ARQ includes all the information necessary to route the call, MVAM sends an ACF message to the gateway. The gateway then processes the call as if the following VoIP parameters were set to these values:

```
vpn-mode=yes
single-dial-enable=yes
```

If MVAM or a third-party billing application used with MultiVoice requires that a caller perform two-stage dialing for the call (dialing the destination telephone number after dialing into the MultiVoice gateway), an admission reject (ARJ) message is issued directing the gateway to set `single-dial-enable=no` for the call. The gateway then resumes call handling as if the call had just arrived from the PSTN, but prompts the caller to enter the destination telephone number (`single-dial-enable=no`) before continuing with call processing.

Static announcement branding

When the multiple logical gateway feature is enabled on a MultiVoice gateway, MVAM or a third-party billing application can select a set of voice announcements for playback from multiple sets of voice announcements stored on the gateway. This feature is known as *branding*.

By sending either an ARJ or ACF message that contains an announcement directory specifier, the gateway plays back voice announcements from the named directory on the PC flash card for the current call.

While running the branding instructions, the gateway searches for the voice announcement directory using the value assigned to the Voice-Ann-Dir parameter. If `voice-ann-dir = /current` (default) and the MVAM requests a specific directory (brand) of announcements for a call, the gateway searches for those announcements starting in the `/current` directory. For example, if MVAM specified `italian`, the gateway searches for announcements in the directory `/current/italian/`.

Note: Lucent Technologies recommends that you use only four brands of static announcements because of limitations in the announcement cache size. Using more than four brands degrades announcement quality and overall gateway performance.

Configurable call timers

TAOS 9.0 supports the use of configurable call timers, controlled by MVAM or a third-party billing application, which support timed billing plans (such as pre-paid telephone cards and pre-paid cellular accounts).

Using an ACF message, MVAM or a third-party billing application, set the following timers:

Timer	Description
Call countdown timer	Sets the time remaining before a gateway disconnects the current call. When this timer expires, the gateway plays an announcement that time has expired and disconnects the call. • By default, once the timer is set on the gateway, the `h323drq.au` announcement file is played back for the caller upon call termination. • If the MVAM or third-party billing application uses its own countdown timer, the announcement specified in an Disengage Request (DRQ) message can be used to select a different announcement file for playback upon call termination.
Call disconnect warning timer	Specifies when a call disconnect warning announcement is played for the caller. This announcement alerts the caller to the time remaining before this call is terminated. • By default, once this timer is set on the gateway, the `h323bkin.au` announcement file is played back for the caller when this timer expires. • If the MVAM or third-party billing application uses its own disconnect warning timer, the announcement specified in an Interrupt Request (IRQ) message can used to select a different announcement file for playback when this timer expires.

H.323 call-specific administration messages

Call-administration information is transmitted as part of the nonstandard data included in registration, admission, and status (RAS) messages exchanged between the gateway and gatekeeper for each call. This data consists of a set of parameters using URL encoding, as described in RFC 1738, with each parameter composed of a set of attribute-value pairs.

This nonstandard data can include the following call administration information:

- ANI/CLID
- Conference identifier
- User PIN
- Inbound or outbound trunk identification
- Information enabling voice announcement playback
- Voice announcement playback
- Internal call timer and disconnect timer settings
- Call failures
- Call results
- Trunk group and DS0 status information
- Available digital signal processors (DSPs)
- Maximum number of calls a MultiVoice gateway can support

Trunk and Call status reporting

Each MultiVoice gateway reports its current call processing status as part of a Registration Request (RRQ) message to MVAM. This message includes data-on-trunk, trunk group, and DS0 status. The initial RRQ message, sent to MVAM when a gateway is initialized, contains a full report on all the trunks used by the physical gateway. The RRQ messages sent during keep-alive registration include only the status changes since the previous registration message.

DS0 Status (in-service/out-of-service)

A MultiVoice gateway reports the following information to MVAM for each trunk:

- Trunk group
- Physical address
- DS0 service status (in-service or out-of-service)

  Note: A DS0 is in service for a logical gateway when it belongs to the associated trunk group and is in the "up" state. Information regarding DS0 activity (in-use, free) is not reported via RRQ. Instead, it is handled separately from per-call trunk and DS0 reporting mentioned. See "Trunk and DS0 selection (per call)" on page 136.

Trunk group and physical address (shelf, slot, an so on) information is provided to MVAM to allow dynamic tracking of DS0 activity and trunk group assignments (and to support DS0 selection by physical-address for outbound PSTN calls in future releases).

Full trunk and DS0 status reporting is performed only when necessary for enhancing gateway performance. Full RRQs are used to report complete trunk and DS0 information, usually when a gateway is initialized or when requested by MVAM. Lightweight RRQs report only status changes for trunk and DS0 information. MVAM can request complete trunk and DS0 information by responding to a lightweight RRQ with a registration reject (RRJ) message containing a reject reason of `FullRegsitrationRequired`.

Note: In the current software release, trunk and DS0 status is not reported for BRI lines. Only the following information is reported for MultiVoice gateways using BRI:

- Number of idle VoIP ports.
- Value of maxCalls in a VoIP profile.

Trunk and DS0 reporting (per call)

For each call processed by a MultiVoice gateway, trunk group and physical address information for the DS0 connection is reported. This information is sent from the gateway to the gatekeeper as nonstandard data in these registration, admission, and status (RAS) messages, for the following call types:

Message	Call type	Trunk or DS0 information
Admission request (ARQ)	Inbound (from PSTN)	The trunk group and physical address of the DS0 upon which the call arrived.
Bandwidth request (BRQ)	Outbound (to PSTN)	The trunk group and physical address of the DS0 upon which the call went out.

Message	Call type	Trunk or DS0 information
Disengage request (DRQ)	Inbound (from PSTN) and Outbound (to PSTN)	The physical address of the DS0. **Note:** For outgoing PSTN calls, the trunk group or DS0 information might not be present.
Disengage confirmation (DCF)	Inbound (from PSTN) and Outbound (to PSTN)	The trunk group and DS0 information for gatekeeper-initiated call terminations.

Trunk and DS0 selection (per call)

Currently, MultiVoice gateways support only trunk-group based routing for outbound PSTN calls. To configure trunk-group based routing, you must enable trunk groups in the System profile of each gateway in the MultiVoice network. Each T1 line must also be assigned a trunk group.

Note: You must assign trunk groups only at the T1 level.

The physical address information collected by the gateway for each DS0 is used currently by MVAM to dynamic track DS0 activity. It is currently not used for DS0-to-DS0 linking.

To support call-specific administration instructions from a MultiVoice Access Manager (MVAM), TAOS 9.0 adds the `Gk-Mlg-Control` parameter in the VoIP profile.

Gk-Mlg-Control

Description: Enables the MultiVoice gateway to accept and process call-specific administration instructions from a MultiVoice Access Manager 3.0 or later release. When enabled, the gateway can apply call-specific processing instructions for PIN authentication, single-stage or two-stage dialing, voice announcement playback, and configuring call timers for pre-paid billing. Values received from MVAM or a third-party billing system override parameter settings in the VoIP profile for processing the current VoIP call.

Rules used for performing call-specific administration are configured on MVAM and are used when partitioning MultiVoice gateways into multiple logical gateways. This method allows the MVAM to administer a single physical gateway as if it were multiple gateways, partitioning the gateway according to trunk groups, DNIS, and time of day.

Usage: Specify `yes` to enable processing of call-specific administration instructions or `no` (default) to revert to global administration of VoIP calls using the values set in the VoIP profile.

Example: `set gk-mlg-control=yes`

Dependencies: This parameter has the following dependencies:

- If `gk-mlg-control=yes`, the value of Vpn-Mode defaults to `N/A.`
- If `gk-mlg-control=yes`, the value of Single-Dial-Enable defaults to `N/A.`
- Changes to this parameter are effective with the next VoIP call.

Location: Voip { X X }

MultiVoice: Report trunk capacity to the gatekeeper

This enhancement enables a MultiVoice gateway to report the following information for all T1 or T3 trunks attached to a MAX TNT unit to the MultiVoice Access Manager (MVAM):

- Trunk status at initialization
- Changes in trunk status after initialization
- Trunk profile changes after initialization
- Trunk group changes after initialization

Changes in trunk availability are reported for trunks used for both VoIP and data calls as part of the nonStandardData byte sent in subsequent registration request (RRQ) messages.

A MAX TNT unit configured as a MultiVoice gateway periodically reports trunk availability to the MVAM in response to an information request (IRQ) message and as part of a subsequent registration request (RRQ) message. The nonStandardData byte uses two data fields for reporting trunk status for all enabled trunks and changes to trunk status. The MVAM extracts this information to determine which trunk groups have available channels for egressing VoIP calls.

When a MAX TNT unit is initialized, it sends a full RRQ to MVAM, reporting the status of all enabled trunks. When trunk group routing is enabled for VoIP calls, this information allows MVAM identify available channels for egressing VoIP calls.

A lightweight RRQ reports only changes to the original trunk information. This message is issued in response to an IRQ message from the MVAM and in a subsequent RRQ message. This message reports the following information:

- Any changes in trunk status (such as when an active trunk goes down or an inactive trunk comes up), as they occur. An RRQ is sent immediately for just the changed trunk, even if that trunk is disabled.
- Any changes made in the profile for a T1 or T3 trunk, (such as signaling mode or default call type), which affect trunk availability. An RRQ is sent immediately for just the profile change, unless the change is to the T1 > Line-Interface > Enabled parameter, which is reported as a change in trunk status, not a profile change.
- Whenever the assigned trunk group usage changes for a trunk or channel. An RRQ is sent immediately for only the enabled trunks. Disabled trunks are *not* included in the report.
- Any changes in channel status (such as when an active trunk goes down or an inactive trunk comes up), as they occur. Every lightweight RRQ automatically reports channel status changes when the status of any channel differs from the previously reported. Only the channel status at the time the RRQ is generated matters. The channel status changes that occur between scheduled keep-alive registrations are not reported.

In the absence of trunk or profile changes, channel status is checked and compared for every lightweight RRQ sent by the MAX TNT unit during keep-alive registration. Disabled trunks are also included in the RRQ sent during keep-alive registration.

E1 CMF R2: Collect 15-digit dial strings

This enhancement increases the maximum number of digits a MAX TNT unit functioning as a MultiVoice gateway can require before accepting an incoming call that uses R2 signaling,

without waiting for end-of-pulsing signaling. A MultiVoice gateway can hold dial strings of up to 15 digits from E1 trunks supporting inband CMF R2.

This feature makes MultiVoice gateways compatible for use on European telephone systems that use E.164 addresses of up to to 15 digits long, without waiting for an end-of-pulse signal. Previously, MultiVoice gateways could be configured to collect dial strings of up to only 11 digits. For European networks using dial strings that were 12 digits or longer, a MultiVoice gateway could only be configured to wait for the end-of-pulse signal to confirm it received all the dialed digits.

To support this feature, TAOS 9.0 adds new values to the E1 line Number-Complete parameter in the Line-Interface subprofile of the E1 profile.

Number-Complete

Description: This parameter is modified to enable detection and collection of up to 15 digits for inbound dialed telephone numbers on MultiVoice gateways using E1 trunks supporting inband CMF R2.

Usage: This parameter now accepts values from `0-digits` through `15-digits`, or `end-of-pulse` as valid entries.

Example: The following example illustrates how to enable multiple logical gateway processing on this MAX TNT:

```
admin> read e1 { 1 1 7 }
E1/{ 1 1 7 } read

admin> set number-complete = 15-digits

admin> write
E1/{ 1 1 7 } written
```

Dependencies: The following dependencies apply to this parameter:

- Changes are applied with the next VoIP call
- This parameter defaults to `N/A` when the Signaling-Mode parameter is assigned the following values:
  - `e1-kuwait-signaling`
  - `isdn`
  - `p7`
  - `dpnss`
  - `none`

Location: E1 { x x x } > Line-Interface

MultiVoice: Delay of charges until call is answered (true connect)

This feature causes a MAX TNT unit acting as a MultiVoice gateway to wait until a VoIP call is answered before alerting the local switched telephone network that a call has been connected.

This enhancement enables true-connect signaling on MultiVoice gateways. A MultiVoice gateway can be configured through the command-line interface (CLI) to delay sending the

connect message to the incoming (ingress) PSTN switch until the following information is received from the outging (egress) MultiVoice gateway:

- An H.323 alerting message
- A call progress message from the outgoing (egress) PSTN indicating the call has been answered

Previously, incoming VoIP calls from the PSTN were connected at the near-end gateway before any H.323 signaling was sent to the far-end gateway. As a result, a PSTN charge was incurred at the time of connection to the near-end gateway, before the called party received and answered the call from the far-end gateway.

The true-connect feature requires a default call type of VoIP on T1 or E1 trunks accepting incoming VoIP calls, as illustrated by the following:

```
[in T1/{ shelf-1 slot-10 1 }:line-interface]
default-call-type = voip
```

```
[in E1/{ shelf-1 slot-11 1 }:line-interface]
default-call-type = voip
```

To enable and disable true-connect signaling for VoIP calls, TAOS 9.0 adds the True-Connect-Enable parameter to the VoIP { x x } profile.

True-Connect-Enable

Description: The True-Connect-Enable parameter is used to enable or disable true-connect signaling for VoIP calls. When enabled, the ingress MultiVoice gateway delays PSTN alerting and sending connect messages to match the equivalent H.323 alerting and connect messages.

Usage: Specify `yes` or `no`. The default is `no`.

Value	Description
`yes`	An alerting message is sent to the ingress PSTN switch only when an H.323 alerting message is received on the ingress VoIP gateway, and a PSTN connect message is sent only when the H.323 VoIP call has been answered. As a result, no charges are incurred for incomplete calls.
`no`	An alerting message is sent to the ingress PSTN switch as soon as the connection is established with the ingress MultiVoice gateway. As a result, the caller incurs a PSTN charge at the time of connection to the near-end gateway, before the called party has received and answered the call from the far-end gateway.

Example: `set true-connect-enable = yes`

Dependencies: The True-Connect-Enable parameter has the following dependencies:

- The Default-Call-Type parameter in the T1 or E1 Line-Interface profile must be set to `voip` for T1 or E1 trunks used for incoming VoIP calls that require true-connect signaling. Setting this parameter to `voip` causes *all* calls received on the trunk to be mapped to VoIP.
- With ISDN trunks, Lucent Technologies recommends that you set the T310 timeout on the telco switch or PBX to 30 seconds or greater when using the true-connect feature. The T310 timeout includes the time that the called party's telephone is ringing, so a 10-second timeout can cause the near-end gateway to tear down the call too soon.

- When the true-connect feature is enabled and a VoIP call fails before the PSTN call is fully connected, the gateway is still able to send an appropriate tone or voice announcement to the caller.

MultiVoice: Support for configurable operator assistance

TAOS 9.0 allows callers to request operator assistance during the dialing phase of a MultiVoice call. A MultiVoice gateway can be assigned a dial string up to 5 digits long, that a caller can enter to connect an operator.

Callers can enter a set of digits (such as `*0` or `09`) when they need operator assistance during the dialing stage of a MultiVoice call. The digit string used to request operator assistance is defined in the VoIP profile.

When the caller enters the operator assistance command, the MAX TNT sends the digits requesting operator assistance to the MultiVoice Access Manager (MVAM). MVAM translates these digits into the telephone number sent to the far-end MultiVoice gateway to connect this caller to an operator telephone.

Once the call is connected, the digit strings used to request operator assistance are available for normal call processing functions, such as responding to automated attendants, AUDIX, and so on.

The operator assistance option is supported for MultiVoice gateways operating as either Multiple Logical Gateways (`gk-mlg-control` = `yes`) or as a single gateway (`gk-mlg-control` = `yes`).

TAOS 9.0 modifies the user interface by adding the Transfer-to-Operator parameter to the VoIP profile.

Transfer-to-Operator

Description: The Transfer-to-Operator parameter is used to define the dial string a caller enters when requesting operator assistance. This parameter can be up to five digits long.

Usage: The Transfer-to-Operator feature is enabled by the entry of a 2-to-5-digit dial string containing an asterisk (*) in either the first or second position. This parameter accepts the asterisk (*) plus any number(s) 0 through 9 as a valid entry. By default this value is `*0`. You disable this feature by assigning a NULL value to the Transfer-to-Operator parameter.

Example: `set transfer-to-Operator = *9`

Dependencies: The Transfer-to-Operator parameter has the following dependencies:

- The first or second digit of the dial string must always be an asterisk (*).
- A MultiVoice gateway must be configured for two-stage dialing (`single-dial-enable` = `no`).
- A translation rule must be defined in one of the ingress translation tables used by MVAM that contains the actual dialed number used to connect calls to operator assistance.

Location: Voip { 0 0 }, Voip { x x }

MultiVoice: Support for sequential call dialing without authentication

Some callers must enter a personal identification number (PIN) to authenticate MultiVoice calls. These callers can now dial subsequent VoIP calls without reentering their PIN, as long as they do not terminate the connection between the PSTN and near-end MAX TNT unit acting as a MultiVoice gateway. MultiVoice users need only authenticate once, for the initial VoIP call, to initiate subsequent calls.

At the end of the initial VoIP call, the caller must remain on the line after the destination telephone is hung up. Once the caller hears the dial tone, the caller can dial another telephone number without entering the PIN.

This feature is supported for MultiVoice gateways operating as either Multiple Logical gateways (`gk-mlg-control` = `yes`) or as a single gateway (`gk-mlg-control` = `yes`).

To support this feature, TAOS 9.0 adds the Sequential-Calls-Enable parameter to the VoIP profile.

Sequential Calls Enable

Description: The Sequential-Calls-Enable parameter is used to enable callers who must enter a PIN to authenticate MultiVoice calls to dial subsequent VoIP calls without reentering their PIN.

Usage: Set the value of the Sequential-Calls-Enable parameter to `yes`, the default, to allow users to authenticate once for their initial VoIP call and then place additional calls without authenticating. To disable the feature, set the value of the Sequential-Calls-Enable parameter to `no`.

Example: `set sequential-calls-enable = yes`

The new value is applied with the next VoIP call received by the MultiVoice gateway.

Dependencies: The Sequential-Calls-Enable parameter has the following dependencies:

- The MAX TNT must be configured for two-stage dialing and PIN collection (`vpn-mode=no`).
- If the original call was an operator-assisted call, the caller is automatically disconnected.
- If the original call used single-stage dialing (not prepaid and within a calling card environment), the caller is automatically disconnected.

Location: VoIP { 0 0 }, VoIP { x x }

MultiVoice: Support for VoIP PSTN attributes

This enhancement changes the way an egress MultiVoice gateway manages call signaling with the switched network. TAOS 9.0 includes the following changes to call signaling:

- Enables transparent delivery of Q.931 or Q.850 cause codes received from the PSTN by the far-end MultiVoice gateway to the near-end MultiVoice gateway.
- Allows configuration of the bearer capabilities sent in the Q.931 setup message by the far-end MultiVoice gateway for outbound calls to the switched network.

- Allows reporting of the Q.931 Progress Indicator information element (IE) in the proceeding and alerting message by the near-end MultiVoice gateway to the switched network.

Changes to MultiVoice call signal processing made in TAOS 9.0 are consistent with the following telecommunications standards:

- ITU Telecommunication sector standard (ITU-T) Q.931, *Digital Subscriber Signalling System No. 1 (DSS 1) - ISDN User-Network Interface Layer 3 Specification for Basic Call Control* (Mar. 1993), International Telecommunications Union
- ITU Telecommunication sector standard (ITU-T) Q.850, *Usage of Cause and Locations in the Digital Subscriber Signalling System No. 1 and the Signalling System No. 7 ISDN User Part* (Mar. 1993), International Telecommunications Union

Transparent reporting of disconnect cause codes

MultiVoice provides transparent reporting of call disconnect cause codes for both the far-end MultiVoice gateway and near-end MultiVoice gateway using Q.931 (H.323) or Q.850 (SS7) signaling.

Transparent reporting is enabled and a VoIP call is disconnected

- If the inbound PSTN connection uses PRI signaling, the near-end MultiVoice gateway passes the Q.931 disconnect message—generated by the far-end PSTN and passed across the packet network by the far-end MultiVoice gateway—directly to the near-end switched network. When the Q.931 disconnect message is received by the local telephone company switch, it plays the appropriate tone for the caller. The near-end MultiVoice gateway does not play any voice announcement or tones.
- If the inbound PSTN connection uses inband signaling, or the call is disconnected internally, the near-end MultiVoice gateway responds to the Q.931/Q.850 cause codes, reporting the information to the MultiVoice Access Manager (MVAM). The near-end MultiVoice gateway then generates either the appropriate call progress tone or voice announcement for the caller, depending upon the instructions it receives from MVAM.

Transparent reporting is disabled and a VoIP call is disconnected

- The near-end MultiVoice gateway plays the appropriate voice announcement or tones for the end user.
- The near-end MultiVoice gateway then sends Q.931 disconnect message with cause code NORMAL (16). to the local telephone company switch.

Configurable bearer capabilities for outbound calls to the PSTN

With TAOS 9.0, you can configure an outbound MultiVoice gateway to request one of the following bearer services from the switched circuit network for outbound VoIP calls for outbound call processing:

- Speech
- Unrestricted digital information
- Restricted digital information
- 3.1-kHz audio
- Video

In previous software releases, the outbound MultiVoice gateway requested only speech bearer service when connecting a VoIP call to the switched telephone network. The outbound MultiVoice gateway sends the bearer service request to the switched telephone network in the bearer service information element of the call setup message. For more information see “4.5.5 Bearer capability” in ITU Telecommunication sector standard (ITU-T) Q.931, *Digital Subscriber Signalling System No. 1 (DSS 1) - ISDN User-Network Interface Layer 3 Specification for Basic Call Control* (Mar. 1993).

Q.931 call signaling progress indicator

With TAOS 9.0, you can configure an outbound MultiVoice gateway to forward the Q.931 progress indicator information element as part of the alerting and proceeding messages sent to the ingress switched network. The Q.931 progress indicator information element describes call routing events on the egress switched telephone networks used for a VoIP call.

When use of the progress indicator information element is enabled, a MultiVoice gateway includes the call routing event descriptions in the alerting and/or proceeding messages sent to the ingress switched network. The progress indicator information element reports one of the following routing descriptions:

- Call is not end-to-end ISDN; further call progress information might be available in-band.
- Destination address is non-ISDN.
- Origination address is non-ISDN.
- Call has returned to the ISDN.
- Interworking has occurred and has resulted in a telecommunication service change.
- Inband information or an appropriate pattern is now available.

For more information on the use of the progress indicator information element, see “4.5.23 Progress Indicator” and “Annex G” in ITU Telecommunication sector standard (ITU-T) Q.931, *Digital Subscriber Signalling (sic) System No. 1 (DSS 1) - ISDN User-Network Interface Layer 3 Specification for Basic Call Control* (Mar. 1993).

Detecting and reporting call progress

To detect progress indicators in call alerting and proceeding messages on a near-end MultiVoice gateway, you must set the the Default-Call-Type parameter (in the Line-Interface subprofile of the T1/E1 profile) to differentiate between inbound VoIP calls from the switched telephone network and non-VoIP calls. For a trunk line using inband signaling, the Default-Call-Type parameter specifies the type of call that the MultiVoice gateway can accept for incoming calls on this trunk, for purposes of call routing. TAOS 9.0 extends the usage and meaning of the Default-Call-Type parameter to both to inband and PRI signaling for VoIP calls. Set `default-call-type` to `voip` to enable sending the progress indicator in call alerting and proceeding messages for PRI signaling.

Pstn-Attribute subprofile

TAOS 9.0 adds the Pstn-Attribute subprofile to the VoIP profile. This profile contains the parameter Cause-Code-Transparency, Alert-Progress-Indicator, Proceed-Progress-Indicator, and Bearer-Capability, which are defined in the following sections.

Following is a sample Pstn-Attribute subprofile, shown with default settings:

```
[in VOIP/{ "" "" }:pstn-attribute (new)]
cause-code-transparency = no
alert-progress-indicator = no-progress-indicator
proceed-progress-indicator = no-progress-indicator
bearer-capability = speech
```

Cause-Code-Transparency

Description: Enables transparent delivery of the Q.931 (H.323 VoIP) or Q.850 (SS7) disconnect cause codes generated by the far-end switched network—passed across the packet network from the far-end MultiVoice gateway to the near-end MultiVoice gateway—to the local telephone company. The local telephone company switch then plays the appropriate tone or disconnect message for the caller.

Usage: Set to `yes` to enable this parameter. By default, this parameter is set to `no`.

Example: `set cause-code-transparency = yes`

Dependencies: Consider the following:

- For callers to hear both a busy signal and the call failure message whenever voice announcement reporting is enabled (`h323-voice-ann-enabled = yes`), you must enable the `cause-code-transparency` parameter. When voice announcements are enabled and the `cause-code-transparency` parameter is disabled, callers do not hear the busy tone. Instead, the near-end MultiVoice gateway plays the call failure message.
- Changes to the Cause-Code-Transparency parameter take effect with the next VoIP call.

Location: Voip { 0 0 } > Pstn-Attribute, Voip { x x } > Pstn-Attribute

Alert-Progress-Indicator

Description: Sets the type of call-progress events that are captured and reported by the MultiVoice gateway in the information element for the Q.931 Alert message progress indicator. Once configured, MultiVoice gateways report when specific call-routing events occur for VoIP calls passing from the packet network and the switched telephone network.

Usage: Specify one of the following values:

Parameter value	Description
`no-progress-indicator`	Disables alert reporting of call routing events on the egress switched telephone network. This is the default setting.
`none-end2end-isdn`	Sets the egress MultiVoice gateway to report when calls are connected to a egress switched telephone network that does not use ISDN signaling. The egress switched telephone network can support robbed-bit or detectable DTMF.
`dest-non-isdn`	Sets the egress MultiVoice gateway to report when calls are connected to a egress switched telephone network that does not use ISDN signaling, such as a transit network or private network, which does not return call progress signals to the MultiVoice gateway.

Parameter value	Description
`orig-non-isdn`	Sets the ingress MultiVoice gateway to report when calls are received from a local switched telephone network that does not use ISDN signaling, such as a transit network or private network, which does not provide call progress signals to the MultiVoice gateway.
`return-to-isdn`	Sets the egress MultiVoice gateway to report when calls connected across a transit network are routed back to a trunk supporting ISDN signaling.
`interworking-occurred`	Sets the egress MultiVoice gateway to report if interworking occurs when a call is connected to the switched telephone network (such as when the selected bearer capability is not supported or when a resource or route with the preferred capability is not available).
`inband-info-available`	Sets the egress MultiVoice gateway to report if inband call progress signaling or other supported non-ISDN signaling is available from the switched telephone network for the connected call.

Example: `set alert-progress-indicator = none-end2end-isdn`

Dependencies: Changes to the Alert-Progress-Indicator parameter take effect with the next VoIP call.

Location: Voip { 0 0 } > Pstn-Attribute, Voip { x x } > Pstn-Attribute

Proceed-Progress-Indicator

Description: Specifies the type of call progress events that the MultiVoice gateway captures and reports in the information element for the Q.931 proceeding message progress indicator. Once configured, MultiVoice gateways report when specific call routing events occur for VoIP calls passing from the packet network and the switched telephone network.

Usage: The Proceed-Progress-Indicator parameter can be assigned the following values:

Parameter value	Description
`no-progress-indicator`	Disables alert reporting of call routing events on the egress switched telephone network. This is the default setting.
`none-end2end-isdn`	Sets the egress MultiVoice gateway to report when calls are connected to a egress switched telephone network that does not use ISDN signaling. The egress switched telephone network can support robbed-bit or detectable DTMF.
`dest-non-isdn`	Sets the egress MultiVoice gateway to report when calls are connected to a egress switched telephone network that does not use ISDN signaling, such as a transit network or private network, which does not return call progress signals to the MultiVoice gateway.

Parameter value	Description
`orig-non-isdn`	Sets the ingress MultiVoice gateway to report when calls are received from a local switched telephone network that does not use ISDN signaling, such as a transit network or private network, which does not provide call progress signals to the MultiVoice gateway.
`return-to-isdn`	Sets the egress MultiVoice gateway to report when calls connected across a transit network are routed back on to trunk supporting ISDN signaling.
`interworking-occurred`	Sets the egress MultiVoice gateway to report when interworking occurs when a call is connected to the switched telephone network (such as when the selected bearer capability is not supported or when a resource or route with the preferred capability is not available).
`inband-info-available`	Sets the egress MultiVoice gateway to report if inband call progress signaling or other supported non-ISDN signaling is available from the switched telephone network for the connected call.

Example: **`set proceed-progress-indicator = none-end2end-isdn`**

Dependencies: Changes to the Proceed-Progress-Indicator parameter take effect with the next VoIP call.

Location: Voip { 0 0 } > Pstn-Attribute, Voip { x x } > Pstn-Attribute

Bearer-Capability

Description: Sets the MultiVoice gateway to request a specific bearer service from the egress switched circuit network for outbound VoIP calls. This request is transmitted to the switched telephone network in the bearer service information element of the call setup message sent by the MultiVoice gateway.

Usage: The Bearer-Capability parameter can be assigned the following values:

Parameter value	Specifies
`speech`	Switched network routing over a channel that supports speech bearer capability (the default setting).
`unrestricted-digital-info`	Switched network routing over a channel that supports unrestricted digital information (UDI) bearer capability.
`restricted-digital-info`	Switched network routing over a channel that supports restricted digital information (RDI) bearer capability.

Parameter value	Specifies
`audio-3100hz`	Switched network routing over a channel that supports digital audio bearer capability up to 3.1 kHz.
`video`	Switched network routing over a channel that supports video signaling bearer capability.

Example: `set bearer-capability = audio-3100hz`

Dependencies: Changes to the Bearer-Capability parameter take effect with the next VoIP call.

Location: Voip { 0 0 } > Pstn-Attribute, Voip { x x } > Pstn-Attribute

MultiVoice: Support for IPDC RMCP and AMCP messages

This enhancement provides support for IP Device Control (IPDC) protocol RMCP and AMCP messages for processing SS7 VoIP calls requests.

The RMCP and AMCP messages are used to modify parameters for RCCP or ACCP controlled (VoIP) calls. With this software version, the messages can be used to modify the following parameters:

- VoIP encoding type (G.711, G.729, and so forth) with Tag `0x70`. Note that this release also introduces support for G.723 (5.4 Kbps) encoding for SS7 VoIP calls. Following are supported values for VoIP encoding:

Encoding type	Value
G.711-U-Law	`0x00`
G.723	`0x04`
G.711-A-Law	`0x08`
G.729	`0x12`

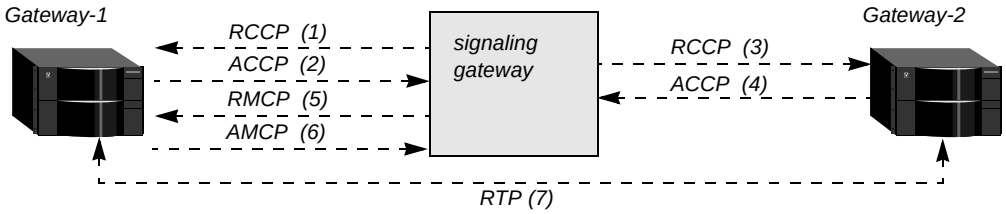
- Packet Loading (frames per packet) with Tag `0x73`. Values depend on the VoIP encoding type.
- Destination Port Type with Tag `0x65`. Note that for IPDC 0.12 VoIP calls, the only supported values for Source (`0x65`) and Dest Port (`0x66`) Type tags are SCN (`0x00`) and RTP (`0x01`) respectively.
- Listen IP address with Tag `0x5D`.
- Listen RTP port with Tag `0x5E`.
- Send IP address with Tag `0x5F`.
- Send RTP port with Tag `0x60`.

The required/optional status of these tags is supplied in “Details of IPDC message support” on page 149.

Send IP address and Send RTP port tags

With RMCP support of Tags `0x5F` and `0x60`, the MAX TNT can now allocate its own listen IP addresses and RTP ports. The exchanges used in this process are shown in Figure 4.

Figure 4. Example IPDC message exchanges



IPDC messages to establish RTP listen addresses and ports are exchanged as follows:

1 The signaling gateway sends an RCCP message to Gateway-1, in which the RTP port is either not specified or is 0, but with no IP address or RTP port tags.

2 Gateway-1 returns its RTP listen IP address and RTP port to the signaling gateway in an ACCP message, using tags `0x5D` and `0x5E`.

3 The signaling gateway sends an RCCP message to Gateway-2, in which the destination listen IP address and destination listen RTP port number obtained from Gateway-1 are specified.

4 Gateway-2 returns its RTP listen IP address and RTP port to the signaling gateway in an ACCP message, using tags `0x5D` and `0x5E`.

5 The signaling gateway sends an RMCP message to Gateway-1, in which the destination listen IP address and destination listen RTP port number obtained from Gateway-2 are specified.

6 Gateway-1 returns an AMCP message to the signaling gateway.

7 RTP communication commences between the Gateway-1 and Gateway-2.

Related routing issues

For all VoIP calls, it is important to avoid routing RTP traffic through the MAX TNT shelf controller. For that reason, when allowing the MAX TNT gateway to allocate its own RTP address, you must set the System-IP-Addr parameter in the IP-Global profile to an interface address other than the default zero address (which defaults to the shelf-controller Ethernet port). For example, the following commands set the system address to the address of a port on an Ethernet card in slot 12:

```
admin> get ip-interface { { 1 12 1 } 0} ip-address
[in IP-INTERFACE/{ { shelf-1 slot-12 1 } 0 }:ip-address]
ip-address = 1.1.1.1/24

admin> read ip-global
IP-GLOBAL read

admin> set system-ip-addr = 1.1.1.1/24

admin> write
IP-GLOBAL written
```

In addition, it is important that VoIP calls can always find a route to the next-hop gateway on the path to the destination VoIP gateway. The route can be learned dynamically or configured

as a static route. Many sites choose to configure default routes for VoIP traffic, so RTP packets will never be dropped due to lack of routing information. For example, the following commands configure a default route named `voip` to a next-hop gateway at 2.2.2.2:

```
admin> new ip-route voip
IP-ROUTE/voip read

admin> set gateway = 2.2.2.2/24

admin> write
IP-ROUTE/voip written
```

Details of IPDC message support

The following IPDC messages are now supported:

- RMCP (`0x0015`)
- AMCP (`0x0016`)

The tags shown in Table 24 are now supported for the RMCP message.

Table 24. RMCP message tags

Tag	Parameter description	Status
`0x65`	Source port type (PSTN only)	Required
`0x07`	Source module number	Required
`0x0D`	Source line number	Required
`0x15`	Source channel number	Required
`0x66`	Destination port type (RTP only)	Required
`0x70`	VoIP encoding type (new G.723 value supported)	Optional
`0x73`	Packet loading (value depends on VoIP encoding type)	Optional
`0x5D`	Destination listen IP address (see Note below)	Optional
`0x5E`	Destination listen RTP port number (see Note below)	Optional
`0x5F`	Destination send IP address (see Note below)	Optional
`0x60`	Destination send RTP port number (see Note below)	Optional

Note: The last four tags in Table 24 are required if values are nonzero. In addition, if an IP address tag is present, the matching port tag must also be present. This requirement also applies to the same tags in AMCP messages, listed in Table 25. In addition, RCCP and ACCP messages have been modified to use the same requirements regarding these tags.

The tags shown in Table 25 are now supported for the AMCP message.

Table 25. AMCP message tags

Tag	Parameter description	Status
`0x65`	Source port type (PSTN only)	Required
`0x07`	Source module number	Required
`0x0D`	Source line number	Required
`0x15`	Source channel number	Required
`0x66`	Destination port type (RTP only)	Required

Table 25. AMCP message tags (continued)

Tag	Parameter description	Status
0x70	VoIP encoding type (new G.723 value supported)	Required
0x73	Packet loading (value depends on VoIP encoding type)	Required
0x5D	Destination listen IP address	Optional
0x5E	Destination listen RTP port number	Optional
0x5F	Destination send IP address	Optional
0x60	Destination send RTP port number	Optional

Note: Tags 0x70 and 0x73 are required in AMCP messages, because one use of RMCP is to query the information for a VoIP call.

Tunneling extension features

Proxy LCP and authentication for L2TP tunnels

If a PPP client's profile is configured to initiate an L2TP tunnel, the MAX TNT unit operating as an L2TP access concentrator (LAC) attempts to open a tunnel or reuse an existing tunnel after initial authentication of the connection.

If the LAC preauthenticates the client's dial-in call by means of calling line ID (CLID) or Dialed Number Information Service (DNIS), it initiates the tunnel to the L2TP network server (LNS), and the LNS begins Link Control Protocol (LCP) negotiation with the mobile client.

If the LAC authenticates the PPP client's dial-in call by means of a name and password, it negotiates LCP with the client and then opens the PPP Auth state. Previously, the information obtained from authentication and LCP negotiation on the LAC was not forwarded, so the LNS had to restart negotiation with the client. Now, the LAC forwards relevant LCP information (*proxy LCP*) and the caller's name and password (*proxy authentication*). This feature provides quicker connection of the client, because the LNS does not need to restart negotiation.

With proxy LCP, instead of sending an empty LCP Config Request packet in the data stream to the LNS, the LAC sends the LNS the following information:

- The first LCP Config Request packet received from the client.
- The last LCP Config Request packet received from the client.
- The last LCP Config Request packet the LAC sent to the client

With proxy authentication, the LAC completes PPP authentication of the dial-in call and then sends the caller's name and password to the LNS in the appropriate L2TP attribute-value pairs.

Proxy LCP and authentication occur for digital calls that are authenticated through any PPP authentication protocol (such as PAP, CHAP, or MS-CHAP) but not for analog PPP connections authenticated by a terminal-server login. For security reasons, the terminal server erases the caller's name and password immediately after authenticating the user.

Support for Tunnel-Client-Auth-ID and Tunnel-Server-Auth-ID

MAX TNT units now support additional tunnel authentication settings to enable more flexible and secure establishment of L2TP and Layer 2 Forwarding (L2F) tunnels. Previously, because of constraints caused by L2TP and RADIUS protocol requirements, tunnel authentication required that every network access server (NAS) in the network used the same system name, even when the network spanned multiple administrative domains.

With the current software version, each NAS can send a unique system name for tunnel authentication purposes. The name can be specified on a per-connection or per-server basis. If RADIUS accounting is enabled, the MAX TNT unit reports the names used for tunnel authentication in the Stop record.

Note: Tunnel authentication occurs before a tunnel is established between two end points. It is negotiated between the MAX TNT unit and a tunnel server and is independent of user authentication. If tunnel authentication fails, all pending calls associated with the tunnel are dropped.

For L2TP tunnels, because the LAC can now specify its name on a per-connection basis, you can configure profiles to create parallel tunnels to the same destination. For example, some sites use parallel tunnels to separate data streams that are directed to the same LNS but destined for different networks.

Overview of local profile settings

The Client-Auth-ID parameter, previously supported only for L2F tunnels, now has an expanded functionality. The Server-Auth-ID parameter is new.

For details about how the system uses these settings to determine whether to use an existing tunnel or start a new one, see “How the system finds a matching tunnel” on page 156. For details about how the system determines which name to use for tunnel authentication, see “How the system name is selected” on page 156.

Following are the parameters (shown with default values) relevant to the enhanced tunnel authentication provided by this feature:

```
[in CONNECTION/"":tunnel-options]
profile-type = disabled
tunneling-protocol = atmp-protocol
primary-tunnel-server = ""
secondary-tunnel-server = ""
password = ""
client-auth-id = ""
server-auth-id = ""

[in TUNNEL-SERVER/""]
shared-secret = ""
client-auth-id = ""
server-auth-id = ""

[in L2-TUNNEL-GLOBAL]
l2tp-system-name = ""
l2f-system-name = ""

[in SYSTEM]
name = ""
```

Parameter	Specifies
Profile-Type	Type of tunneling profile. Must be set to `mobile-client` for PPP clients using L2TP or L2F tunneling.
Tunneling-Protocol	Protocol used to establish the tunnel. Must be set to `l2f-protocol` or `l2tp-protocol` to use this feature.
Primary-Tunnel-Server	IP address or hostname of a tunnel end point. If a DNS lookup returns several IP addresses, the system attempts to establish a tunnel to each address in turn. If the primary server is unavailable, the system attempts to establish a tunnel to the secondary server.
Secondary-Tunnel-Server	IP address or hostname of a tunnel end point. If a DNS lookup returns several IP addresses, the system attempts to establish a tunnel to each address in turn. If the primary server is unavailable, the system attempts to establish a tunnel to the secondary server.
Password	Password used for authenticating the tunnel.
Client-Auth-ID	Name sent to the tunnel server for authenticating the tunnel. The name can contain up to 31 characters. Note that L2F does not support Client-Auth-ID or Server-Auth-ID from a Tunnel-Server profile. For more details, see “How the system name is selected” on page 156.
Server-Auth-ID	Name sent from the tunnel server to the system initiating the tunnel during the tunnel authentication phase. The name can contain up to 31 characters. Note that this field is currently ignored if it is specified in a Connection profile. Note that L2F does not support Client-Auth-ID or Server-Auth-ID from a Tunnel-Server profile.
Shared-Secret	Shared secret for authenticating the tunnel.
L2TP-System-Name L2F-System-Name	Name sent to the tunnel server for authenticating the tunnel if Client-Auth-ID is not specified. See “How the system name is selected” on page 156.
Name	Name sent to the tunnel server for authenticating the tunnel if Client-Auth-ID is not specified, and L2TP-System-Name or L2F-System-Name is not specified. See “How the system name is selected” on page 156. If the domain name is configured in the IP-Global profile, the specified system name is concatenated with the domain name.

Overview of RADIUS attribute-value pairs

RADIUS supports this feature by using the following attribute-value pairs. These attribute-value pairs support tag fields, as described in the RFC 2868. Each tag value (from 1 to 31) defines an independent tunnel attempt description. The Tunnel-Client-Auth-ID and Tunnel-Server-Auth-ID attributes can be specified in Access-Response packets and are generated in Accounting-Request packets.

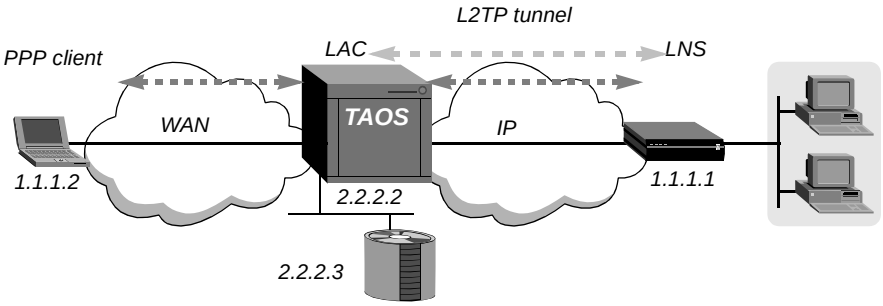
RADIUS attribute	Value
Tunnel-Type (64)	Tunneling protocol(s) to be used. Must be set to L2TP (3) or L2F (2) to use this feature.

RADIUS attribute	Value
Tunnel-Server-Endpoint (67)	IP address or hostname of the tunnel end point. If a DNS lookup returns several IP addresses, the system attempts to establish a tunnel to each address in turn.
Tunnel-Password (69)	Shared secret for authenticating the tunnel.
Tunnel-Client-Auth-ID (90)	Name sent to the tunnel end point by the system requesting the tunnel (the NAS or LAC) during the tunnel authentication phase. The name can contain up to 31 characters. See “How the system name is selected” on page 156.
Tunnel-Server-Auth-ID (91)	Name sent from the tunnel end point (the gateway or LNS) to the system initiating the tunnel during the tunnel authentication phase. The name can contain up to 31 characters. This attribute does not apply unless the protocol used to establish the tunnel is L2TP or L2F. The attribute can be specified in Access-Response packets and is generated in Accounting-Request packets.

Examples of tunnel authentication

In the example shown in Figure 5, a PPP client dials into a MAX TNT unit to tunnel into its home network across the Internet.

Figure 5. Example of L2TP tunnel authentication



For the purposes of this example, the MAX TNT unit authenticates the initial PPP dial-in by its Dial Number Information Service (DNIS) number. (DNIS authentication is not required for tunnel authentication.) The MAX TNT unit operates as an L2TP access concentrator (LAC), but not as an LNS. So this example shows only the LAC configuration.

System configuration

For RADIUS to authenticate callers, the External-Auth profile must be configured. For example, the following commands configure the MAX TNT unit to use a RADIUS server for both authentication and accounting purposes:

```
admin> new external-auth
EXTERNAL-AUTH read

admin> set auth-type = radius

admin> set acct-type = radius

admin> set rad-auth-client auth-server-1 = 2.2.2.3
```

```
admin> set rad-auth-client auth-port = 1645
admin> set rad-auth-client auth-key = key
admin> set rad-auth-client auth-timeout = 5
admin> set rad-acct-client acct-server-1 = 2.2.2.3
admin> set rad-acct-client acct-port = 1646
admin> set rad-acct-client acct-key = key
admin> set rad-acct-client acct-timeout = 5
admin> write
EXTERNAL-AUTH written
```

The next set of commands configures the system to collect DNIS information:

```
admin> read answer-defaults
ANSWER-DEFAULTS read
admin> set clid-auth-mode = dnis-first
admin> set ppp-answer receive-auth-mode = any-ppp-auth
admin> write -f
ANSWER-DEFAULTS written
```

Connection-based tunnel authentication

The following commands configure a local Connection profile for the PPP client and specify a Client-Auth-ID name:

```
admin> new connection dnis-user
CONNECTION/dnis-user read
admin> set calledNumber = 001
admin> set tunnel-options profile-type = mobile-client
admin> set tunnel-options tunneling-protocol = l2tp-protocol
admin> set tunnel-options primary-tunnel-server = 1.1.1.1
admin> set tunnel-options password = conn-pass
admin> set tunnel-options client-auth-id = conn-LAC
admin> write -f
CONNECTION/dnis-user written
```

Note that you need not assign an IP address because it is assigned by the LNS. Following is a comparable RADIUS profile:

```
001 User-Password = "Ascend-DNIS", Service-Type = Dialout
   Tunnel-Type = L2TP,
   Tunnel-Server-Endpoint = 1.1.1.1,
   Tunnel-Password = conn-pass,
   Tunnel-Client-Auth-ID = conn-LAC
```

With the sample profiles, the LAC uses DNIS to authenticate the PPP client's dial-in call. It then initiates a tunnel to the LNS if a tunnel does not already exist to that end-point address. When the MAX TNT unit requests the tunnel, it passes the LNS the string `conn-LAC` as its local system name, and uses `conn-pass` as the password to authenticate the tunnel. The LNS uses the same strings to authenticate the LAC before establishing the tunnel.

Server-based tunnel authentication

The following commands configure a local Connection profile for the PPP client and do not specify a password or Client-Auth-ID name:

```
admin> new connection dnis-user
CONNECTION/dnis-user read

admin> set calledNumber = 001

admin> set tunnel-options profile-type = mobile-client

admin> set tunnel-options tunneling-protocol = l2tp-protocol

admin> set tunnel-options primary-tunnel-server = lns.example.com

admin> write -f
CONNECTION/dnis-user written
```

Following is a comparable RADIUS profile:

```
001 User-Password = "Ascend-DNIS", Service-Type = Dialout
   Tunnel-Type = L2TP,
   Tunnel-Server-Endpoint = lns.example.com,
```

With the sample profiles, the LAC uses DNIS to authenticate the PPP client's dial-in call. It then initiates a tunnel to the LNS if a tunnel does not already exist to that end-point address. If tunnel authentication is enabled and no tunnel password is specified in the Connection profile, the unit searches for a Tunnel-Server profile before requesting the tunnel. If it finds a Tunnel-Server profile for the LNS, the unit sends the Client-Auth-ID to the LNS and the end points use the tunnel password (the shared secret) to authenticate the tunnel. Following is a sample Tunnel-Server profile that specifies a password and local system name for use in tunnel authentication:

```
admin> new tunnel-server lns.example.com
TUNNEL-SERVER/lns.example.com read

admin> set shared-secret = ts-pass,

admin> set client-auth-id = ts-LAC

admin> write
TUNNEL-SERVER/lns.example.com written
```

Following is a comparable RADIUS profile:

```
lns.example.com Password = "", Service-Type = Dialout
   Tunnel-Password = ts-pass
   Tunnel-Client-Auth-ID = ts-LAC
```

Note: If no Tunnel-Server profile exists, the LAC proceeds as described in "How the system name is selected" on page 156.

Examples of creating parallel L2TP tunnels to the same end point

After the LAC has authenticated a PPP client's dial-in call, it looks for an existing tunnel that matches both the tunnel server end point and the Client-Auth-ID specified in the client's profile. If the LAC finds an established tunnel that matches these values, it uses the tunnel. If it does not find a matching tunnel, it initiates a tunnel request. This process can be used to create parallel L2TP tunnels by specifying different Client-Auth-ID values in profiles.

How the system finds a matching tunnel

If the client's profile specifies a hostname as the tunnel-server end point, the system must match both the hostname and the server's actual IP address to allow the client to use an established tunnel.

If Client-Auth-ID is specified in the caller's profile, the system attempts to match the caller to an existing tunnel by using the following values:

- Tunnel server's IP address (and hostname, if specified)
- Client-Auth-ID

If Client-Auth-ID is *not* specified in the caller's profile, the system attempts to match the caller to an existing tunnel by using only the tunnel server's IP address (and hostname, if specified).

If the system finds a match on the basis of the values, it uses the tunnel. If the MAX TNT unit does not find a matching tunnel entry, it initiates a new tunnel request.

How the system name is selected

If tunnel authentication is enabled and the MAX TNT unit is requesting a new tunnel, it looks for a system name to send to the LNS as follows:

1 If available, uses the Client-Auth-ID specified in the caller's Connection profile. If Client-Auth-ID is not specified in the Connection profile, the system goes on to the next alternative.

2 If available, uses the Client-Auth-ID specified in the Tunnel-Server profile for the LNS. If Client-Auth-ID is not specified in a Tunnel-Server profile, the system goes on to the next alternative.

3 If available, uses the L2TP-System-Name specified in the L2-Tunnel-Global profile. If L2TP-System-Name is not specified in that profile, the system goes on to the next alternative.

4 If available, uses the Name specified in the unit's System profile. If Name is not specified in that profile, the system goes on to the next alternative.

5 Sends the string `noname`.

Examples of how Client-Auth-ID settings create parallel tunnels

In this example, the LNS system's DNS hostname is `a.example.com` (a fully qualified domain name), which resolves to two IP addresses, 1.1.1.1 and 1.1.1.2. The hostname `b.example.net` also resolves to the 1.1.1.1 address. Table 26 shows existing tunnels to the LNS, which were authenticated using different Client-Auth-ID strings.

Table 26. Existing tunnels to the same LNS

Address	Client-Auth-ID	Tunnel-Server	Tunnel-ID
1.1.1.1	a1	`a.example.com`	102
1.1.1.1	a2	`a.example.com`	103

Table 27 shows how the system matches the values in the clients' profiles as it receives incoming calls, and the resulting action the system takes in terms of using an existing tunnel or creating a new one.

Table 27. Tunnels created for incoming callers based on profile settings

Values used to match tunnel:			Resulting action	Tunnel-ID
Address	**Client-Auth-ID**	**Tunnel-Server**		
1.1.1.1	a1	`a.example.com`	Reuse tunnel	102
1.1.1.1	a2	`a.example.com`	Reuse tunnel	103
1.1.1.1	b	`b.example.net`	Establish new tunnel	104
1.1.1.1	b	`a.example.com`	Establish new tunnel	105
1.1.1.1		`a.example.com`	Reuse tunnel	102 or 103
1.1.1.1	a2	`b.example.net`	Establish new tunnel	106
1.1.1.2	a1	`a.example.com`	Establish new tunnel	107

Note: The caller that does not supply a Client-Auth-ID string matches the tunnel-server end point, so the existing tunnel to that end point (Tunnel-ID 102) is reused.

Examples of configuration errors causing multiple tunnels

Configuration errors can lead to unintentional parallel tunnels to the same tunnel end point. For this reason, you should either use the Client-Auth-ID setting for all user profiles to a particular LNS or decide *not* to use that setting for callers tunneling to that LNS.

For example, suppose your RADIUS users file contains the following two user profiles and tunnel server profile:

```
user1 Password = userpass
   Tunnel-Type = L2TP,
   Tunnel-Server-Endpoint = lns.example.com,
   Tunnel-Client-Auth-ID = A-LAC,
   ...
user2 Password = userpass
   Tunnel-Type = L2TP,
   Tunnel-Server-Endpoint = lns.example.com,
   ...
lns.example.com User-Password = "", Service-Type = Dialout
   Tunnel-Password = tunpass,
   Tunnel-Client-Auth-ID = AllMyLACs
```

If `user1` calls in first and establishes a tunnel, `user2` can reuse that tunnel, as shown in Table 28.

Table 28. Tunnels created when user1 dials in first (configuration error not detected)

Values used to match tunnels:			Resulting action	Tunnel-ID
Address	**Client-Auth-ID**	**Tunnel-Server**		
2.2.2.2	A-LAC	lns.example.com	Create new tunnel	88
2.2.2.2		lns.example.com	Reuse tunnel	88

However, if `user2` calls in first and establishes a tunnel, the system will obtain a system name for authentication from the tunnel-server profile. When `user1` dials in, the caller will be unable to reuse the tunnel, because the authentication names will not match. This situation is shown in Table 29.

Table 29. Tunnels created when user2 dials in first (configuration error shown)

Values used to match tunnels:			Resulting action	Tunnel-ID
Address	**Client-Auth-ID**	**Tunnel-Server**		
2.2.2.2	AllMyLACs	lns.example.com	Create new tunnel	40
2.2.2.2	A-LAC	lns.example.com	Create new tunnel	42

Support for tunnel assignment IDs

Like the Client-Auth-ID described in “Support for Tunnel-Client-Auth-ID and Tunnel-Server-Auth-ID” on page 151, the tunnel assignment ID provides a mechanism for informing the LAC whether to assign a client session to an existing tunnel or to create a new one. It is also used for grouping client sessions into specific tunnels. For details, see RFC 2868.

How tunnel assignment IDs affect tunnel matching

In addition to the criteria described in “How the system finds a matching tunnel” on page 156, this feature enables the system to perform an additional, final check for a tunnel assignment ID when selecting an existing tunnel or deciding to create a new one. After comparing the tunnel transport address and, if specified in the client’s profile, the tunnel server’s hostname (Server-Endpoint) against existing tunnels, the system begins comparing the following optional parameters, in the order shown:

- Client-Auth-ID specified in the client’s profile and the Client-Auth-ID used for existing tunnels
- Assignment-ID specified in the client’s profile and the tunnel assignment ID of existing tunnels

The client profile matches existing tunnels only if both the Client-Auth-ID and the tunnel assignment ID match. A null value in any one of these fields in an existing tunnel matches only a null value in the corresponding parameter in the client profile. If the MAX TNT unit does not find a matching tunnel entry, it initiates a new tunnel request.

Overview of local profile settings

The following new parameters (shown with default values) have been added to local profiles:

```
[CONNECTION/"":tunnel-options]
assignment-id = ""
```

Parameter	Specifies
Assignment-ID	Identification (name) assigned to tunnels to allow grouping sessions. A text string of up to 31 characters. The value has local significance only. It is not transmitted to the remote tunnel end point.

Overview of RADIUS attribute-value pair

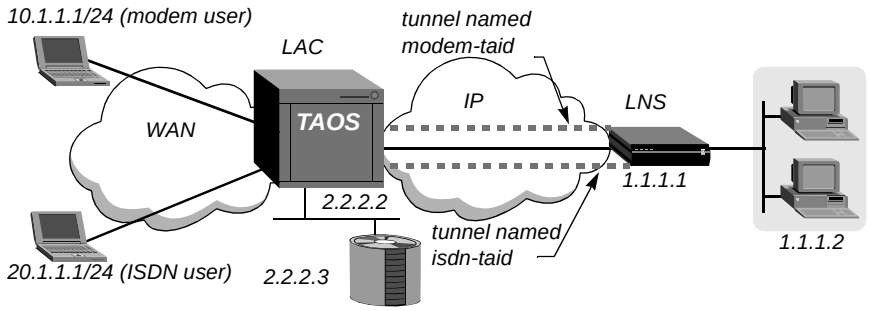
RADIUS supports this feature by means of the following attribute-value pair:

RADIUS attribute	Value
Tunnel-Assignment-ID (82)	Identification (name) assigned to tunnels to allow grouping sessions. A text string of up to 31 characters. The value has local significance only. It is not transmitted to the remote tunnel end point.

Example of configuring a tunnel assignment ID

In this example, the MAX TNT unit is configured to perform tunnel authentication for L2TP tunnels. The two PPP clients shown in Figure 6 are configured to use different tunnels to the LNS on the basis of their tunnel assignment IDs. (The same clients could be configured to use the same multiplexed tunnel if their tunnel assignment IDs were set to the same string.)

Figure 6. L2TP tunnel setup that uses tunnel assignment IDs



The following set of commands enables tunnel authentication on the MAX TNT unit:

```
admin> read l2-tunnel-global
L2-TUNNEL-GLOBAL read

admin> set l2tp-mode = lac

admin> set l2tp-auth-enabled = yes

admin> write
L2-TUNNEL-GLOBAL written
```

The following set of commands creates local Connection profiles for the two mobile clients:

```
admin> new connection modemuser
CONNECTION/modemuser read
admin> set ppp-options recv-password = test
admin> set tunnel-options profile-type = mobile-client
admin> set tunnel-options tunneling-protocol = l2tp-protocol
admin> set tunnel-options primary-tunnel-server = 1.1.1.1
admin> set tunnel-options password = shared
admin> set tunnel-options client-auth-id = taos-unit
admin> set tunnel-options assignment-id = modem-taid
admin> write
CONNECTION/modemuser written
admin> new connection isdnuser
CONNECTION/isdnuser read
admin> set ppp-options recv-password = test
admin> set tunnel-options profile-type = mobile-client
admin> set tunnel-options tunneling-protocol = l2tp-protocol
admin> set tunnel-options primary-tunnel-server = 1.1.1.1
admin> set tunnel-options password = shared
admin> set tunnel-options client-auth-id = taos-unit
admin> set tunnel-options assignment-id = isdn-taid
admin> write
CONNECTION/isdnuser written
```

Following are comparable RADIUS profiles:

```
modemuser Password = "test"
    User-Service = Framed-User,
    Framed-Protocol = PPP,
    Test-Idle-Limit = 0,
    Tunnel-Type = L2TP :1,
    Tunnel-Server-Endpoint = 1.1.1.1 :1,
    Tunnel-Client-Auth-ID = taos-unit: 1,
    Tunnel-Password = shared,
    Tunnel-Assignment-ID = modem-taid:1

isdnuser  Password = "test"
    User-Service = Framed-User,
    Framed-Protocol = PPP,
    Test-Idle-Limit = 0,
    Tunnel-Type = L2TP :1,
    Tunnel-Server-Endpoint = 1.1.1.1 :1,
    Tunnel-Client-Auth-ID = taos-unit: 1,
    Tunnel-Password = shared,
    Tunnel-Assignment-ID = isdn-taid:1
```

RADIUS accounting support

RADIUS accounting Stop records display the Tunnel-Assignment-ID used for the user-session. For example:

```
Tue May 2 15:58:08 2000
        User-Name = "modemuser"
        NAS-Identifier = 2.2.2.2
        NAS-Port = 11313
        NAS-Port-Type = Async
        Acct-Status-Type = Stop
        Acct-Delay-Time = 0
        Acct-Session-Id = "317658341"
        Acct-Authentic = Local
        Acct-Session-Time = 112
        Acct-Input-Octets = 2155
        Acct-Output-Octets = 513
        Acct-Input-Packets = 23
        Acct-Output-Packets = 14
        Ascend-Disconnect-Cause = 185
        Ascend-Connect-Progress = 60
        Ascend-Xmit-Rate = 28800
        Ascend-Data-Rate = 33600
        Ascend-PreSession-Time = 19
        Ascend-Pre-Input-Octets = 0
        Ascend-Pre-Output-Octets = 0
        Ascend-Pre-Input-Packets = 0
        Ascend-Pre-Output-Packets = 0
        Ascend-Modem-PortNo = 1
        Ascend-Modem-SlotNo = 7
        Ascend-Modem-ShelfNo = 1
        Caller-Id = "1119855510"
        Client-Port-DNIS = "3826"
        Tunnel-Type = L2TP
        Tunnel-Server-Endpoint = "1.1.1.1"
        Tunnel-Client-Auth-ID = "taos-unit"
        Tunnel-Server-Auth-ID = "max6k-lns"
        Tunnel-Assignment-ID = "modem-taid"
```

Support for L2TP hidden attributes

By supporting hidden attributes, this release completes the process of bringing the L2TP implementation on the MAX TNT unit into conformance with Draft 16 of RFC 2661, *Layer Two Tunneling Protocol "L2TP"*. The unit now correctly parses and decrypts hidden attributes as well as the random vector attribute-value pair, with the exception that the assigned tunnel ID attribute-value pair cannot be hidden in the Start-Control-Connection-Request (SCCRQ) message. The unit does not hide any attributes except under the control of a debug flag.

Support for Tunnel-Private-Group-ID in L2TP

The Tunnel-Private-Group-ID parameter identifies a gateway connection to an L2TP network server (LNS), MAX TNT units configured as LACs, and non-Lucent L2TP network servers that support this feature.

This feature is already supported in the Ascend Tunnel Management Protocol (ATMP). Currently, Lucent LNSs do not support this feature.

The RADIUS `Tunnel-Private-Group-ID` attribute and the TAOS `homenetworkname` parameter in the `tunnel-options` subprofile of the local connection profile now apply to L2TP tunnels (LAC only) as well as ATMP tunnels.

Example of a RADIUS profile:

```
MobileClient Password = "Ascend",
   Tunnel-Type = L2TP,
   Tunnel-Private-Group-ID = RadiusPrivateGroup,
   Tunnel-Server-Endpoint = 21.21.21.21
```

Enhanced RADIUS accounting and call logging for VPNs

This release introduces two new attributes for RADIUS accounting and call logging with virtual private network (VPN) configurations. Two other VPN-specific attributes, formerly used only for authentication, now can be used for RADIUS accounting and call logging as well.

The following attributes have been added for RADIUS accounting and call logging with VPN configurations:

- Acct-Tunnel-Connection (68)
- Ascend-Tunnel-Auth-Type (260)

Following are the attributes that were formerly used only for authentication, and which now appear in RADIUS accounting and call logging packets:

- Ascend-Tunnel-VRouter-Name (31)
- Ascend-Vrouter-Name (102)

The description of each new attribute related to RADIUS accounting and call logging with VPNs follows. For information on the existing Ascend-Tunnel-VRouter-Name and Ascend-Vrouter-Name attributes, see the *TAOS RADIUS Guide and Reference*.

Acct-Tunnel-Connection (68)

Description: An RFC standard attribute that identifies the tunnel session for an L2TP tunnel.

Usage: Acct-Tunnel-Connection appears in Accounting-Request and Account-Response packets. Along with the Tunnel-Client-Endpoint and Tunnel-Server-Endpoint attributes, Acct-Tunnel-Connection uniquely identifies a session. The Acct-Tunnel-Connection string appears in the following format:

ipaddr1-tunnel1-session1-ipaddr2-tunnel2-session2

Element	Description
ipaddr1, ipaddr2	IP addresses of the tunnel end points.
tunnel1	Tunnel ID (in hexadecimal) for the tunnel end point at *ipaddr1*.
tunnel2	Tunnel ID (in hexadecimal) for the tunnel end point at *ipaddr2*.
session1	Session ID (in hexadecimal) for the tunnel end point at *ipaddr1*.
session2	Session ID (in hexadecimal) for the tunnel end point at *ipaddr2*.

Example: `acct-tunnel-connection = "170.20.200.2-0001-001D-200.168.24.141-0005-005F"`

Dependencies: To simplify the matching of accounting records at both ends of the tunnel, the numerical value of *ipaddr1* must be less than that of *ipaddr2.*

See Also: Ascend-Tunnel-Auth-Type, Ascend-Tunnel-VRouter-Name, Ascend-Vrouter-Name, Tunnel-Client-Endpoint, Tunnel-Server-Endpoint in the *TAOS RADIUS Guide and Reference*

Ascend-Tunnel-Auth-Type (260)

Description: A 16-bit vendor-specific attribute (VSA) that describes the method used to authenticate the call.

Usage: Specify one of the following values:

- Textual (1) specifies a username/password exchange.
- Chap (2) specifies Challenge Handshake Authentication Protocol (2).
- Pap (3) specifies Password Authentication Protocol (3).
- None (4) specifies no authentication.
- MsChap (5) specifies Microsoft CHAP.

Example: `Ascend-Tunnel-Auth-Type = Chap`

Dependencies: Ascend-Tunnel-Auth-Type is sent with a tag value of 0 (zero).

See Also: Acct-Tunnel-Connection, Ascend-Tunnel-VRouter-Name, Ascend-Vrouter-Name, Tunnel-Client-Endpoint, Tunnel-Server-Endpoint in the *TAOS RADIUS Guide and Reference.*

Support for L2TP, PPTP, and L2F disconnect and progress codes

This release supports a new set of disconnect and progress codes for the following tunneling protocols:

- Layer 2 Tunneling Protocol (L2TP)
- Point-to-Point Tunneling Protocol (PPTP)
- Layer 2 Forwarding (L2F)

You can use the disconnect and progress codes introduced in this release to troubleshoot the configuration of your unit's tunneling systems. These progress and disconnect codes are logged by the Syslog, RADIUS accounting, and call logging servers.

A tunnel entry or tunnel descriptor refers to the set of values that describes a tunnel end point. For example, the following RADIUS attributes constitute a tunnel entry for the user `JohnDoe`:

```
JohnDoe Password="abc"

   Tunnel-Type = L2TP,
   Tunnel-Server-Endpoint = LNS.example.com,
   Tunnel-Password = difficult,
   Tunnel-Client-Auth-ID = MyLAC,
   Tunnel-Server-Auth-ID = HisLNS ...
```

The following progress codes are now reported by tunneled calls using L2TP, PPTP, or L2F:

Progress code	Description
240	Tunnel is being started. Set when the unit has determined that a call must be tunneled. Errors occurring during this period usually indicate that a tunnel server entry is invalid.
241	System is resolving the address of a remote tunnel server end point by DNS. Errors occurring during this period usually indicate a problem with DNS or an invalid tunnel server entry.
242	System is contacting a remote tunnel server. Set after the unit has resolved the address of the remote tunnel end point and has started trying to contact it. Errors occurring during this phase usually indicate that the remote server is unreachable because it is not operating, because no route to it exists, or because of tunnel authentication errors, depending on the tunneling protocol used. Call authentication errors do not usually affect this phase.
243	Call is being transferred to a remote tunnel server. Set after the unit contacts the remote tunnel end point. At this point, the two tunnel end points are actively working on establishing the tunnel, authenticating each other if needed, and negotiating the tunnel session (tunnel authentication is independent of user authentication). Errors occurring during this phase usually indicate resource or configuration problems on the remote server, such as incorrect or invalid tunnel passwords or configuration conflicts.
244	Tunnel is established. Call has been tunneled to the remote tunnel end point and it is ready to transfer data. This code is sometimes superseded by code 60 (Session up).

The following disconnect codes are now supported by tunneled calls using L2TP:

Disconnect code	Description
730	Unknown reason.
731	Tunnel protocol is disabled by a configuration setting or software license.
732	Operation is disallowed by configuration.
733	Invalid tunnel end point entry values.
734	Out of resources.
735	Tunnel end point is being shut down. All calls and tunnels using the same tunnel end point are affected.
736	Administrative tunnel disconnect. All calls using the same tunnel are affected. Other tunnels to the same tunnel end point are not affected.
750	Server is not responding because it timed out.
751	Server is not responding to periodic hello commands.
752	Tunnel authentication failed.

753	Missing tunnel password. A tunnel password is required but cannot be found.
754	Tunnel protocol error. A protocol mismatch probably occurred between the tunnel end points.
770	Call was cleared due to carrier loss.
771	Call failed because no carrier was detected.
772	Call failed due to a busy signal.
773	Call failed due to a lack of dial tone.
774	Call failed due to an invalid destination number.
775	Call failed because of invalid framing or because no framing was detected.
776	Incoming call was rejected by the remote tunnel end point for unspecified reasons.
777	Outgoing call was rejected by the remote tunnel end point (LAC) for unspecified reasons.
778	Call was not established within the allotted time.

For example, if you issue the `l2tpstop` command on the LNS terminal screen to disconnect an L2TP tunnel, the following codes are be reported on the LAC and LNS:

```
admin> l2tpstop 2

LAC: c=736 p=244 (DIS_L2TUNNEL_ADMIN_DISCONNECT, PR_TUNNEL_UP)
   Syslog :
      May  9 16:21:52 max ASCEND: call 2 CL 0K u=test c=736 p=244
      s=14400 r=14400

    RADIUS accounting:
      Ascend-Disconnect-Cause = 736 (DIS_L2TUNNEL_ADMIN_DISCONNECT)
      Ascend-Connect-Progress = 244 (PR_TUNNEL_UP)

LNS: c=736 p=60 (DIS_L2TUNNEL_ADMIN_DISCONNECT, PR_LAN_SESSION_UP)
   Syslog :
      May  9 16:21:51 maxlns ASCEND: call 3 CL 0K u=test c=736 p=60
      s=14400 r=14400

   RADIUS accounting:
      Ascend-Disconnect-Cause = 736 (DIS_L2TUNNEL_ADMIN_DISCONNECT)
      Ascend-Connect-Progress = 60 (PR_LAN_SESSION_UP)
```

New ATMP disconnect codes

This release includes several new Ascend Tunnel Management Protocol (ATMP) disconnect codes.

Background

If an ATMP client disconnects because of an ATMP error, the new ATMP disconnect codes can help you diagnose the exact cause of the problem. Each code can appear in a Syslog record or as the value of Ascend-Disconnect-Cause (195) in a RADIUS accounting record.

List of disconnect codes

Following are the new disconnect codes:

Number	Explanation
700	Authentication of the Foreign Agent failed.
701	Tunneling is not enabled on the Home Agent.
702	The system is out of resources because too many tunnels have been established.
703	One of the fields in the TUNNEL message contained an invalid value.
704	The tunnel number in the Generic Routing Encapsulation (GRE) packet is invalid or does not exist. This error usually indicates that one side was reset.
705	The peer agent did not respond.
706	The Connection profile for the Home Network in gateway mode is not active.
707	A Domain Name System (DNS) lookup of the Home Agent could not be resolved to an IP address.
708	This code denotes a general error, and has been superseded by codes 709 through 712. Code 708 appears only if you connect to a unit running software issued before the addition of codes 709 through 712.
709	The Home Agent is not in gateway mode.
710	The Home Agent failed to set up a route.
711	The Foreign Agent detected an idle tunnel and cleared it.
712	The Home Agent detected an idle tunnel and cleared it.

Virtual router extension features

Support for VRouters with ATMP

You can now use multiple virtual routers (VRouters) with the Ascend Tunnel Management Protocol (ATMP).

When a user connects to the MAX TNT, the ATMP tunnel is established by means of the specified VRouter. In previous software releases, you could use only the main VRouter to create an ATMP tunnel. You can assign the same IP address to multiple Home Agents in different private segments and select which Home Agent to use by defining the VRouter in the user's TAOS Connection profile or the user's RADIUS profile.

To specify the VRouter for the Connection profile `commgroup`, proceed as in the following example:

```
[in CONNECTION/commgroup]
admin> set vrouter = companyvr
admin> write
CONNECTION/commgroup written
```

Following is a sample RADIUS profile:

```
commgroup Password="123"
   User-Service=Framed-User,
   Framed-IP-Address=199.199.199.200,
   Framed-IP-Netmask=255.255.255.0,
   Framed-Protocol=PPP,
   Ascend-Route-IP=1,
   Tunnel-Type = ATMP
   Tunnel-Server-Endpoint=10.5.7.2,
   Tunnel-Password = "atmp",
   Ascend-Home-Agent-UDP-Port=5150,
   Ascend-Home-Network-Name="HOMNET",
   Ascend-Tunnel-VRouter-Name = "companyvr"
```

Support for VRouters in IPX networks

TAOS 9.0 release enhances the MAX TNT implementation of virtual routers (VRouters) by adding support for IPX networks. In previous software releases, VRouters were available only for IP networks.

A VRouter is a grouping of the LAN or WAN interfaces in a MAX TNT unit. Each VRouter has its own associated IPX ARP table, IPX Routing table, IPX service table, IPX session table, IPX address pools, IPX ping statistics, IPX traffic statistics, and IPX dial-in route tables. For a discussion of VRouters, see the *APX 8000/MAX TNT/DSLTNT WAN, Routing, and Tunneling Configuration Guide.*

Creating an IPX VRouter

You create a VRouter profile for an IPX network just as you would for an IP network. For detailed instructions on how to create a VRouter profile, see the chapter on virtual routers in the *APX 8000/MAX TNT/DSLTNT WAN, Routing, and Tunneling Configuration Guide*.

To support IPX networks, the VRouter profile now contains the new parameters `ipx-routing-enabled` and `ipx-dialin-pool`. You configure the VRouter profile by setting the following parameters. (Parameters in the VRouter profile that are not in the following list do not apply.)

Parameter	Specifies
Name	Unique name for the VRouter, up to 23 characters. To group interfaces belonging to this VRouter, you specify this value in the IPX-Interface profile. See “Assigning the IPX interface to a VRouter” on page 168.
Ipx-Routing-Enabled	Enable/disable IPX routing on the specified VRouter. Set to `yes` to enable IPX routing on the VRouter. By default, this parameter is set to `no`.
Ipx-Dialin-Pool	Dial-in pool of IPX network addresses to be shared by the IPX WAN interfaces. Specify the addresses in dotted-hexadecimal notation, similar to that of an IPX network number. If no dial-in pool is specified, the MAX TNT uses the global VRouter pool specified in the IPX-GLOBAL profile.

Example of defining an IPX VRouter

The following commands create a VRouter profile ipxcorp1:

```
admin> new vrouter ipxcorp1
VROUTER/ipxcorp1 read

admin> set ipx-routing-enabled = yes
admin> set ipx-dialin-pool = 00:00:00:00

admin> write

VROUTER/ipxcorp1 written
```

Defining a global IPX VRouter

You configure a global VRouter for the MAX TNT unit by setting the `global-vrouter` parameter in the IPX-GLOBAL profile.

Parameter	Specifies
Global-Vrouter	Unique name for a global VRouter, up to 23 characters. By default, this parameter is set to `main`.

The following commands define a global VRouter for the MAX TNT unit in an IPX network:

```
admin> read IPX-GLOBAL
IPX-GLOBAL read

admin> list

[in IPX-GLOBAL]

interface-address = { { any-shelf any-slot 0 } 0 }
ipx-routing-enabled = no
ipx-dialin-pool = 00:00:00:00
global-vrouter = main

admin> set global-vrouter = mainv

admin> write

IPX-GLOBAL written
```

Assigning the IPX interface to a VRouter

You assign an IPX interface to a VRouter by setting the VRouter parameter in the IPX-Interface profile.

Parameter	Description
Vrouter	Assigns the IPX interface to a VRouter. If no VRouter is specified, the interface belongs to the global VRouter.

The following set of commands assigns an IPX interface to the VRouter `ipxcorp1:`

```
admin> read ipx-interface {{1 15 1}}
IPX-INTERFACE/{ { shelf-1 slot-15 1 } 0 } read

admin> list

[in IPX-INTERFACE/{ { shelf-1 slot-15 1 } 0 }]

admin> set vrouter = ipxcorp1

admin> write
```

```
IPX-INTERFACE/{ { shelf-1 slot-15 1 } 0 } written
```

Static routes for VRouters

In an IPX network, the MAX TNT unit uses the VRouter setting specified in the IPX-Route profile to dial out to reach the destination network. You do not have to modify the settings in the IPX-Route profile.

Netware command support for VRouters

The `netware` command now supports the VRouter feature for IPX networks.

Netware

Description: Shows IPX network and server information for a specified VRouter. If no VRouter is specified, the unit displays statistics for the global VRouter.

Usage: `netware [`*`vroutername`*`] [-`*`option`*`]`

Syntax element	Description
`Vroutername`	VRouter for which you want to display IPX network and server information.
`-n`	Displays NetWare IPX networks.
`-p`	Displays NetWare IPX pings.
`-s`	Displays NetWare IPX servers.
`-t`	Displays NetWare IPX statistics.

Current limitations on VRouters in IPX networks

- SNMP management does not display information about the MAX TNT unit on a per-VRouter basis. Errors and events are not logged on a VRouter basis. The existing VRouter implementation in TAOS does not have a MIB.
- The MAX TNT implementation of VRouters for IPX networks does not include support for ATMP or L2TP tunnel handling on a VRouter basis.
- The Service Advertising Protocol (SAP) home server proxy is not handled on a VRouter basis.
- IPX stacking on a VRouter basis is not supported.

Short-duration transaction network extension feature

SDTN support for TPDU terminals

The MAX TNT unit can now provide transport protocol data unit (TPDU) terminal access to short-duration transaction networks (SDTNs). For a discussion about SDTNs, see the *APX 8000/MAX TNT/DSLTNT WAN, Routing, and Tunneling Configuration Guide*. A TPDU terminal uses the HDLC-Normal Response Mode (HDLC-NRM) protocol to access an SDTN as follows:

- TPDU terminals use the all-stations HDLC-NRM address as the secondary address.
- The MAX TNT unit polls the TPDU terminals using its secondary address at link startup.

The value written to this poll address is the value used in the initial Set Normal Response Mode (SNRM) request from the MAX TNT unit. Thereafter, the HDLC-NRM protocol implementation uses the address returned by the secondary station. The default for the station poll address is `255` (All-stations).

You configure the MAX TNT unit by setting `station-poll-address` parameter in the `hdlc-nrm-options` subprofile of the Connection profile. By default, this parameter is set to `255`.

Station-Poll-Address

Description: Specifies the address used by a MAX TNT unit in an HDLC-NRM-SNRM request to poll a secondary transport protocol data unit (TPDU) station in a short-duration transaction network (SDTN).

Usage: Specify an integer from 0 through 255. The default is 255, which is the all-stations address.

Example: `set station-poll-address = 255`

Dependencies: Consider the following:

- For HDLC-NRM support, `encapsulation-protocol` must be set to `hdlc-nrm` and `sdtn-packets-server` must be set to `yes` in the Connection profile.

Location: Connection>HDLC-NRM-Options

IP fax extension feature

Support for the Atlas redialer and DID

TAOS 9.0 enhances IP fax functionality by adding support for the Atlas redialer. This release also introduces support for Direct Inward Dialing (DID) with inbound IP fax calls. For a detailed discussion of the MAX TNT IP fax feature and for information on how to configure your MAX TNT unit for IP fax, see the *APX 8000/DSLTNT/MAX TNT WAN, Routing, and Tunneling Guide*.

Specifying the type of redialer

You can now select the type of redialer for incoming fax calls by setting the new parameter, `dial-type`, in the Ip-Fax profile. In previous software releases, the MAX TNT unit supported only the Mitel redialer. The following example shows how to specify the type of redialer that is being used for an inbound IP fax call:

```
[in IP-FAX]

admin> set dialer-type = atlas

admin> write
IP-FAX written
```

Parameter	Specifies
Dialer-Type	Type of redialer that the MAX TNT unit uses for incoming fax calls. Specify `mitel` (the default) for the MITEL redialer or `atlas` for the Atlas redialer.

Support for DID on inbound IP fax calls

This release also enables DID support for inbound IP fax calls on the MAX TNT unit. Every DID subscriber, such as a network user or network printer receives a DID number. To send a fax to a network user or device, senders dial the fax subscriber's DID number and are connected to a TAOS unit.

When a TAOS unit detects an incoming fax call, it authenticates the call by matching the DID number received from the DID trunk against the DID numbers set in the `fax-did` parameter of the Ip-Fax profile. If the numbers match, the TAOS unit initiates a connection with the fax server by sending an incoming fax authentication packet (IFAP) to the fax server for authentication. The incoming fax authentication packet includes the following information:

- Line identifier
- DID number
- Caller ID (if available)

In response to the incoming fax authentication packet, the fax server sends a fax connection response packet (FCRP) that contains one of the following:

- FCRP-NACK—The fax server is unable to handle the call.
- FCRP-ACK—The fax server is unable to handle the call.

After successful establishing a connection with the fax server, the TAOS unit forwards the fax to the fax server.

If the first server fails to accept the call, the MAX TNT unit attempts a connection with the next fax server and so forth. After a connection has been established with a fax server, the MAX TNT unit continues to use that particular fax server for subsequent calls until the connection to that fax server fails. The TAOS unit then attempts to connect to the next fax server specified in the `fax-server` parameter.

The following parameters have been added to the Ip-Fax profile to support DID:

Parameter	**Specifies**
All-Calls-Are-Fax	Enable/disable the handling of all calls as fax calls. Otherwise, the MAX TNT unit authenticates the incoming call based on DNIS or DID. The following values are valid: • `yes`—The MAX TNT unit receives all calls as fax calls. • `no`—The MAX TNT unit authenticates the call based on DID numbers or DNIS numbers, depending the value specified in the `fax-incoming-call-type` parameter.
Fax-Incoming-Call-Type	Type of fax call that the MAX TNT unit will accept. The following values are valid: • `redialer`—All fax calls are redialer calls (the default). • `did`—The MAX TNT unit authenticates the call based on DID entries.
Fax-DID	List of up to eight DID numbers for authentication. A DID number is a dialable string of up to 24 numbers.

Table 30 summarizes how the MAX TNT unit authenticates a call, based on the settings in the `all-calls-are-fax` and `fax-incoming-call-type` parameters.

Table 30. How IP-Fax settings determine authentication

`all-calls-are-fax`	`fax-incoming-call-type`	MAX TNT unit behavior
`yes`	`redialer`	Receives all incoming calls as redialer type of fax call.
`yes`	`did`	Treats all incoming calls as DID type fax calls.
`no`	`did`	Authenticates the call against the DID numbers in the `fax-did` parameter.
`no`	`redialer`	Authenticates the call against the DNIS numbers in the `fax-Dnis` parameter.

Sample IP-fax configuration

Following is a sample IP-Fax profile that configures a MAX TNT unit to use an Atlas redialer for redialing calls and to authenticate calls by means of DID numbers:

```
[in IP-FAX]

ip-fax-enabled = yes
outgoing-fax-port = 10002
server-login = ipfax
dialer-type = atlas
server-password = works
incoming-fax-port = 10002
all-calls-are-fax = no
fax-incoming-call-type = did
fax-dnis = [ 8057 8052 8004 "" "" "" "" "" ]
fax-did  = [ 7470000 7775555 "" "" "" "" "" ]
fax-servers = [ 10.40.40.126 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 ]
```