SIP Working Group
Internet Draft
Document: <draft-dcsgroup-sip-arch-06.txt>
Category: Informational

W. Marshall
AT&T

K. Ramakrishnan
TeraOptic Networks

E. Miller
Terayon

G. Russell
M. Osman
CableLabs

B. Beser
Juniper Networks

M. Mannette
K. Steinbrenner
3Com

D. Oran
F. Andreasen
Cisco

J. Pickens
Com21

P. Lalwaney
Nokia

J. Fellows
Copper Mountain Networks

D. Evans
D. R. Evans Consulting

K. Kelly
NetSpeak

February 28, 2002

Architectural Considerations for Providing Carrier Class Telephony Services Utilizing SIP-based Distributed Call Control Mechanisms

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026[1].

1. Abstract

This document provides an overview of a SIP-based Distributed Call Signaling (DCS) architecture to support carrier class packet-based voice, video, and other real time multimedia services. Companion documents address a specific set of SIP 2.0 protocol extensions and usage rules that are necessary to implement the DCS architecture in an interoperable fashion.

The DCS architecture takes advantage of endpoint intelligence in supporting telephony services without sacrificing the network's ability to provide value through mechanisms such as resource management, lookup of directory information and translation databases, routing services, security, and privacy enforcement. At the same time, the architecture provides flexibility to allow evolution in the services that may be provided by endpoints and the network.

DCS also takes into account the need to manage access to network resources and account for resource usage. The SIP usage rules defined in the accompanying IDs specifically address the coordination between Distributed Call Signaling and dynamic quality of service control mechanisms for managing resources over the access network. In addition, the DCS architecture defines the interaction needed between network provided call controllers, known as a "DCS-

proxy" for supporting these services.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [2].

3. Table of Contents

4. Introduction

This document provides an overview of a SIP-based Distributed Call Signaling (DCS) architecture to support carrier class packet-based voice, video, and other real time multimedia services. The DCS architecture and the corresponding SIP protocol enhancements (described in companion documents) are being developed as part of the cable industry's PacketCable initiative, managed out of CableLabs (see www.cablelabs.com). PacketCable is defining a series of interface specifications that will enable vendors to develop interoperable products for providing internet telephony and other multimedia services over DOCSIS-enabled cable data networks.
The DCS architecture described herein has its roots in the DOSA work performed by AT&T Laboratories ["Distributed Open Signaling Architecture"; Kalmanek, Marshall, Mishra, Nortz, Ramakrishnan, et al.; October, 1998]. A relatively large group of vendors have cooperated in an intensive effort to develop the DCS architecture and SIP protocol extensions described here and in the accompanying protocol drafts. Although DCS was originally designed with cable access networks in mind, the SIP signaling enhancements have general applicability to carrier class VOIP services running over QoS enabled IP networks.

The authors are submitting this draft to the IETF in order to provide general information regarding the DCS architecture and to convey the motivation behind the SIP enhancements recommended in the accompanying protocol drafts. We believe that incorporation of the concepts and mechanisms described in this set of drafts by the IETF into the SIP standard will significantly enhance SIP's ability to function as a carrier-class signaling protocol. Such an enhancement to SIP would undoubtedly aid in its widespread acceptance and deployment. We have incorporated several useful comments received at the IETF SIP Working group on earlier versions of this and the other DCS related drafts.

The PacketCable Draft Specification for DCS is available from the CableLabs website at: ftp://ftp.cablelabs.com/pub/ietfdocs/dcsdraft2.pdf.

4.1 Background and Motivation

The design of the Distributed Call Signaling (DCS) architecture recognizes the trend towards use of packet networks as the underlying framework for communications. These networks will provide a broad range of services, including traditional best-effort data service as well as enhanced, value-added services, such as telephony. At the same time, improvements in silicon will reinforce the trend towards increased functionality in endpoints. These intelligent endpoints will take advantage of the widespread availability of packet networks to enable a rich set of applications and services for users.

However, when the network is used for real-time telephony applications, it is essential to have service differentiation at the IP layer. The ability to control and monitor usage is needed for the provider to be able to provide service differentiation and to derive revenue from the enhanced services. At the same time, the availability of best effort communications and the migration of functionality to the endpoints pose a challenge to the provider to find incentives for users to use or pay for enhanced services.

We see three key functions that a provider can offer, as incentives to use enhanced services. First, the network service provider has the unique ability to manage and provide network layer quality of service. When users depend on the quality of the service, as with telephony, there is a strong incentive to use the enhanced service, rather than a best effort service. Second, the network service provider can play an important role as a trusted intermediary. This includes ensuring the integrity of call routing, as well as ensuring both the accuracy and the privacy of information that is exchanged. The service provider can also add value by ensuring that services are provided consistently and reliably, even when an endpoint is unavailable. Finally, there are a number of services that may be offered more efficiently by the network service provider rather than in endpoints. For example, conference bridging may be more cost effective to implement in a multi-point bridge rather than in every endpoint attached to the network.

A key contribution of the DCS architecture is a recognition of the need for coordination between call signaling, which controls access to telephony specific services, and resource management, which controls access to network-layer resources. This coordination is designed to meet the user expectations and human factors associated with telephony.   For example, the called party should not be alerted until the resources necessary to complete the call are available. If resources were not available when the called party picked up, the user would experience a call defect.   In addition, users expect to be charged for service only after the called party answers the phone. As a result, usage accounting starts only after the called party picks up. Coordination between call signaling and resource management is also needed to prevent fraud and theft of service. The coordination between DCS and Dynamic QoS protocols ensures that users are authenticated and authorized before receiving access to the enhanced QoS associated with the telephony service.

It is important to be able to deploy a residential telephone service at very large scale, cost-effectively.   To achieve this, DCS minimizes the messaging overhead on network call servers, and does not require these servers to maintain call state for active calls. Once a call is established, call state is maintained only where it is needed, in keeping (informally) with the principle of "fate-sharing" at the endpoints that are involved in the call, and at the Edge Routers in the bearer path that are providing differentiated service to the media flow. This allows the network call servers to

scale to support more users, and imposes less stringent reliability requirements on those servers.

DCS is also designed so that calling users receive consistent service even when a called endpoint is unavailable. For example, when an endpoint is unavailable service logic in a network call server can forward telephone calls to a voice mailbox.

4.2 Requirements And Design Principles

In this section, we briefly describe the application requirements that led to a set of DCS signaling design principles. In its most basic implementation, DCS supports a residential telephone service comparable to the local telephone services offered today. In addition to the commonly used service features that need to be supported, there are important requirements in the areas of reliability, performance, and scalability that influence the signaling architecture. Supporting an IP telephony service comparable to the telephony service offered today requires enhanced bearer channel and signaling performance, including:

. Low delay - end-to-end packet delay must be small enough that it does not interfere with normal voice conversations. The ITU recommends no greater than 300 ms roundtrip delay for telephony service.

. Low packet loss - packet loss must be small enough to not perceptibly impede voice quality or performance of fax and voice band modems.

. Short post-dial delay - the delay between the user dialing the last digit and receiving positive confirmation from the network must be short enough that users do not perceive a difference with post-dial delay in the circuit switched network or believe that the network has failed.

. Short post pickup delay - the delay between a user picking up a ringing phone and the voice path being cut through must be short enough so that the "hello" from either the initiator or the receiver of the call is not clipped.

We identify a number of key design principles that arise from the requirements and philosophy outlined above.

1. Providing differentiated network-layer quality of service is essential, while allowing the provider to derive revenues from the use of such differentiated services.

2. The architecture should allow, and even encourage, implementation of services and features in the intelligent endpoints, where economically feasible, while still retaining value in the network and network-based services

3. The architecture must ensure that the network is protected from fraud and theft of service. The service provider must be able to authenticate users requesting service and ensure that only those authorized to receive a particular service be able to obtain it.

4. The architecture must enable the service provider to add value by supporting the functions of a trusted intermediary. This includes

protecting the privacy of calling and called party information, and ensuring the accuracy of the information that is provided in messages from the network.

5. The architecture must enable the service provider to give a consistent view of basic services and features even when customer premise equipment is unavailable, and allow users to take advantage of functionality that is provided in the network, when it is cost-effective and easy to use.

6. The architecture must be implementable, cost-effectively, at very large scale.

4.3 Distributed Call Signaling Architecture

The Distributed Call Signaling Architecture follows the principles outlined above to support a robust telephony service. Figure 1 introduces the key components in the architecture.

The architecture assumes a broad range of DCS-compliant endpoints that provide telephony service to the user including Media Terminal Adapters (MTAs) that may be integrated with a Cable Modem or is a standalone device, as well as other endpoints such as personal computers. The access network interfaces to an IP backbone through a system we refer to as the Edge Router (ER). The ER is the first trusted element within the provider's network and is considered to be the edge of the network for providing access to differentiated quality of service. We believe that the access network is likely to manage resources on a per-flow basis, with associated signaling mechanisms (such as RSVP). The ER performs resource management, acts as a policy enforcement point and as a source of billing information.

DCS-proxies (DPs) process call signaling messages and support number translation, call routing, feature support and admission control. In the context of SIP, a DCS-proxy is a SIP proxy that is involved in processing and forwarding of SIP requests. DPs act as trusted decision points for controlling when resources are committed to particular users. Media servers represent network-based components that operate on media flows to support the service. Media servers perform audio bridging, play terminating announcements, provide interactive voice response services, etc. Finally, PSTN gateways interface to the Public Switched Telephone Network.

```
             +-----+
             | MTA |               MTA: media terminal adapter
             +-----+
                |                  ER: Edge Router
 Access Network |
                |
             +----+
             | ER |
             +----+
                |    +-----------+
                |----| DCS Proxy |
                |    +-----------+
                |
                |    +------------+
Backbone Network |----|Media Server|
                |    +------------+
                |
                |    +------------+
                |----|PSTN Gateway|
                |    +------------+
             +----+
             | ER |
             +----+
                |
 Access Network |
                |
             +-----+
             | MTA |
             +-----+
```

Figure 1: A Simple view of Components of an IP Telephony Architecture used in a HFC Cable Environment.

Telephony endpoints are considered to be "clients" of the telephony service. Consistent with the design principles, the architecture allows a range of services to be implemented by intelligent endpoints. They collect dialed digits, participate in signaling and contain the service logic required for basic call setup and feature support. Endpoints also participate in end-to-end capability negotiation. However, endpoints are not trusted to provide accurate information to the network or to keep information that is received private, except when it is in the endpoint's best interests to do so.

Access to network resources on a differentiated basis is likely to be controlled by the service provider. The ER receives resource management requests from endpoints, and is responsible for ensuring that packets are provided the QoS they are authorized to receive (either through packet marking, or through routing and queueing the packets as a specific QoS assured flow). The ER requires authorization from a network entity (on a call-by-call basis for the

telephony service) before providing access to enhanced QoS for an end-to-end IP flow. The obvious point where this policy and control function resides is the DCS-proxy (also called a gate-controller, because of this responsibility for managing access to enhanced QoS). Thus, the ER is able to ensure that enhanced QoS is only provided for end-to-end flows that have been authorized and for which usage accounting is being done. Since the ER knows about the resource usage associated with individual IP flows, it generates the usage events that allow a user to be charged for service.

We introduce the concept of a "gate" in the ER, which manages access to enhanced quality of service. The gate is a packet classifier and policer that ensures that only those IP flows that have been authorized by the DCS-proxy are granted access to enhanced QoS in the access and backbone networks. Gates are "opened" selectively for a flow. For the telephony service, they are opened for individual calls. Opening a gate involves an admission control check that is performed when a resource management request is received from the endpoint for an individual call, and it may involve resource reservation in the network for the call if necessary. The packet filter in the gate allows a flow of packets to receive enhanced QoS for a call from a specific IP source address and port number to a specific IP destination address and port number.

The DCS-proxy, in addition to implementing many of the call control functions, is responsible for the policy decision regarding whether the gate should be opened. DCS sets up a gate in advance of a resource management message. This allows the policy function, which is at the DCS-proxy, to be "stateless" in that it does not need to know the state of calls that are already in progress.

DCS-proxies are typically organized in domains. A DCS-proxy is responsible for a set of endpoints and the associated ERs. While endpoints are not trusted, there is a trust relationship between the ER and its associated DCS-proxy, since the DCS-proxy plays a role as a policy server controlling when the ER can provide enhanced QoS service. There is also a trust relationship among DCS-proxies. Details of the security model are outside the scope of this draft.

The DCS-proxy is designed as a simple transaction server, so that the failure of a DCS-proxy does not affect calls in progress. A domain will likely have a primary and one or more secondary DCS-proxies. If the primary DCS-proxy fails, only calls in a transient state are affected. The endpoints involved in those calls will time out and retry. All active calls are unaffected. This is possible because the DCS-proxy retains no call state for stable calls. We believe this design makes the DCS-proxy efficient and highly scalable, and keeps the reliability requirements manageable.

DCS supports inter-working with the circuit switched telephone network through PSTN gateways. A PSTN gateway may be realized as a combination of a media controller, media gateway, and a signaling

gateway. A media gateway acts as the IP peer of an endpoint for media packets, converting between the data format used over the IP network and the PCM format required for transmission over the PSTN. The signaling gateway acts as the IP peer of an endpoint for signaling packets, providing signaling inter-working between DCS and conventional telephony signaling protocols such as ISUP/SS7. A media gateway control protocol is used to control the operation of the media gateway from the signaling gateway.

There are additional system elements that may be involved in providing the telephony service. For example, the DCS-proxy may interface with other servers that implement the authorization or translation functions. Similarly, three way calling may be supported using media servers in the network.

4.4 Trust Boundary

The DCS architecture defines a trust boundary around the various systems and servers that are owned, operated by, and/or controlled by the service provider. These trusted systems include the proxies and various servers such as bridge servers, voicemail servers, announcement servers, etc. Outside of the trust boundary lie the customer premises equipment, and various media servers operated by third-party service providers.

Certain subscriber-specific information, such as billing and accounting information, stays within the trust boundary. Other subscriber-specific information, such as endpoint identity, may be presented to untrusted endpoints or may be withheld based on subscriber profiles.

The SIP User Agent (UA) may be either within the trust boundary (e.g. PSTN gateway) or outside the trust boundary (e.g. MTA), depending on exactly what function is being performed and exactly how it is being performed. Accordingly, the procedures followed by a User Agent, as contained in the accompanying drafts, are different depending on whether the UA is within the trust boundary or outside the trust boundary. A trusted user agent is, in almost all cases, equivalent to the combination of an untrusted user agent and a proxy.

4.5 Basic Call Flow

Figure 2 presents a high-level overview of a basic MTA-to-MTA call flow in DCS. Each MTA is associated with a DCS-proxy, which acts as a SIP proxy. When a user goes off-hook and dials a telephone number, the originating MTA (MTA-o) collects the dialed digits and sends the initial INVITE message in SIP, to the "originating" DCS-proxy (DP-o). This INVITE contains SDP proposing a set of codecs that are acceptable to MTA-o (and their implied bandwidth requirements), and an indication of the (mandatory) QoS preconditions [9] needed for the session. DP-o verifies that MTA-o is a valid subscriber of the telephony service (using authentication

information in the INVITE message) and determines whether this subscriber is authorized to place this call. DP-o then translates the dialed number into the address of a "terminating" DCS-proxy (DP-t) and forwards the INVITE message to it.

We assume that the originating and terminating DCS-proxies trust each other. DP-o augments the INVITE message that it forwards with additional information, such as billing information containing the account number of the caller. DP-t then translates the dialed number into the address of the terminating MTA (MTA-t) and forwards the INVITE message to MTA to notify it about the incoming call.

The initial INVITE message invokes call feature handling at the terminating MTA, such as call-forwarding. Assuming that the call is not forwarded, MTA-t negotiates the coding style and bandwidth requirements for the media streams. A reliable provisional 1xx response to the initial INVITE is forwarded back through the DCS-proxies.

```
   MTA-o      ER-o       DP-o       DP-t       ER-t      MTA-t

    | Invite   |          |          |          |          |
    |----------|--------->| Invite   |          |          |
    |          |          |--------->|          |          |
    |          |          |          |  Invite  |          |
    |          |          |          |----------|--------->|
    |          |          |          |          |  183 SDP |
    |          |          |          |<---------|----------|
    |          |          |          |          |          |
    |          |  Gate    |  183 SDP |Gate Setup|          |
    |          |  Setup   |<---------|--------->|          |
    |          |<---------|          |          |          |
    |  183 SDP |          |          |          |          |
    |<---------|----------|          |          |          |
    |          |          | PRACK    |          |          |
    |----------|----------|----------|----------|--------->|
    |          |   200 OK (acknowledging PRACK) |          |
    |<---------|----------|----------|----------|----------|
    |          |          |          |          |          |
    |<---------|--------Reserve Resources-------|--------->|
    |          |          |          |          |          |
    |          |             COMET              |          |
```

```
     |----------|--------- |----------|----------|--------->|
     |               200 OK (acknowledging COMET)           |
     |<---------|----------|----------|----------|----------|
     |          |          |          |          | 180 Ring |
     |          |          | 180 Ring |<---------|----------|
     |          | 180 Ring |<---------|          |          |
     |<---------|----------|          |          |          |
     |          |          | PRACK    |          |          |
     |----------|----------|----------|----------|--------->|
     |          |   200 OK (acknowledging PRACK) |          |
     |<---------|----------|----------|----------|----------|
     |          |          |          |          |          | User
     |          |          |          |          |  200 OK  | Answers
     |          |          |  200 OK  |<---------|----------|
     |          |  200 OK  |<---------|          |          |
     |<---------|----------|          |          |          |
     |   ACK    |          |          |          |          |
     |----------|----------|----------|----------|--------->|
     |          |          |          |          |          |
     |<---------|----------Active  Call----------|--------->|
     |          |          |          |          |          | User
     |          |          |          |          |   BYE    | Hangs Up
     |<---------|----------|----------|----------|----------|
     |          |          |          |          |  200 OK  |
     |----------|----------|----------|----------|--------->|
```

Figure 2: A Basic Call Flow, including Resource Management functions

In the figure, MTA-t sends a 183 SDP message[8] to DP-t. The 183 SDP contains a subset of the capabilities in the INVITE message that are acceptable to MTA-t. The SDP also carries the QoS preconditions from the INVITE. DP-t sends a GATE-SETUP message to the terminating ER (ER-t), conveying policy instructions allowing ER-t to open a gate for the IP flow associated with this phone call. The GATE_SETUP message contains billing information containing the account number of the subscriber that will pay for the call.

DP-t forwards the 183 SDP to DP-o. DP-o sends a GATE-SETUP message to the originating ER (ER-o) to indicate that it can open a gate for the IP flow associated with the phone call. Finally, DP-o forwards 200 OK to MTA-o. The initial INVITE request and 183 SDP response contain a SIP Contact header to indicate the IP address of the remote MTA to be used for subsequent end-to-end SIP signaling exchanges. MTA-o acknowledges the 183 SDP by sending a PRACK [7] directly to MTA-t. The PRACK may contain the SDP to allow for a further step in the negotiation of capabilities for the session.

Once the initial INVITE/183/PRACK exchange has completed, both MTAs reserve the resources that will be needed for the media streams. Once MTA-o has successfully made its reservation, it sends a COMET message [9] to MTA-t, which is immediately acknowledged by MTA-t with a 200-OK. MTA-o uses the COMET message to communicate the fact that the desired pre-conditions necessary for the session as perceived by MTA-o are satisfied (e.g., successful reservation of resources, as perceived by MTA-o.) MTA-t acknowledges the COMET message with a 200 OK final response directly to MTA-o. However, resource reservation from MTA-t's perspective may not be completed yet. Thus, the 200 OK acknowledging the COMET message does not indicate successful resource reservation. Once MTA-t successfully reserves the resources needed for the call, it sends a 180 Ringing through the proxies to indicate that the phone is ringing, and that the calling party should be given a ringback call progress tone. We have not described, in detail, the messaging involved in resource reservation here, as we believe that it is appropriate to allow for a variety of resource management mechanisms. Thus, the MTA may use the resource management mechanism that is most suitable to the network segment that it is attached to. When the called party answers, by going off-hook, MTA-t sends a 200 OK final response through the proxies, which MTA-o acknowledges with an end-to-end ACK. At this point the resources that were previously reserved are committed to this conversation, and the call is "cut through."

Either party can terminate the call. An MTA that detects an on-hook sends a SIP BYE message to the remote MTA, which is acknowledged.

5. Resource Management

DCS's resource management protocols distinguish between two phases: a "Reserve" phase and a "Commit" phase. During the Reserve phase, resources are reserved but are not yet made available to the endpoint. This ensures that resources are available before ringing

the far-end telephone. The Commit phase, which commits the resources associated with the flow, is initiated after ringing the far end telephone and after the called party picks up. At this point, the resources are made available to the endpoint, and recording is started so that the user can be billed for usage. The use of a two-phase approach is essential because of the unique requirements associated with human communication, such as telephony. Recognition of the need for a two phase resource management approach is a significant motivation for the call flow adopted in the previous section.

Although we believe that issues of billing ought not to be the primary consideration in the design of the protocol, the protocol design should not preclude the possibility of usage sensitive billing. Therefore, in addition to ensuring that resources are available before ringing the phone, the two-phase resource management protocol also allows us to preserve the semantics of billing that users are accustomed to, whereby usage recording is not started until the called party picks up the phone. Backbone resources are reserved and allocated in the first phase of the two-phase resource reservation protocol. This is important in order to limit the impact of backbone resource management on post-pickup delay (this minimizes the likelihood of clipping the first few syllables of the conversation).

6. Distributed Call State

In order to provide enhanced services to millions of endpoints, we need an architecture that can be implemented cost-effectively at very large scale. Just as we enable flexibility by exploiting intelligence at the endpoints, services can be provided in a scaleable manner by storing the state associated with applications at the endpoints, rather than in network servers. Especially with telephony, endpoints are directly involved in handling calls and therefore need to maintain and use call state. In contrast, while network servers may need to be involved when setting up a call to gain access to enhanced QoS, there is no fundamental need for those servers to be involved throughout the lifetime of the call. Maintaining state for every call at network servers, while achievable, increases the reliability requirements and load on the servers. The less state kept in the network, the better.

As a result, the DCS-proxies in DCS are designed to be Call stateless transaction servers. The proxy maintains SIP transaction state. So, when a DCS-proxy processes a service request from an endpoint, it maintains state until the transaction is complete, but does not maintain any per-call state about active calls in the network. There are two major advantages to this design. First the reliability of the service does not depend on the reliability of an individual DCS-proxy. A DCS-proxy can fail without affecting calls that are currently in progress. Second, it removes many complex synchronization problems where two (or more) entities need to have simultaneously accurate information. Since interactions with the

DCS-proxies are simple stateless transactions, it is not necessary for consecutive calls to be processed by the same DCS-proxy. DCS-proxy crashes affect only the transient calls (the calls that are in the process of being set up), and not stable conversations. Further, it is likely that most calls in a transient state can be recovered and successfully established through a backup or spare DCS-proxy using endpoint retransmission, with no explicit synchronization protocol required between the DCS-proxies. We believe this design principle will enable us to operate in very large scale, cost effectively. Furthermore it places the function of managing the state of a call where it belongs - at the endpoint. An existing call can only be affected by failures along the path or by failure of the endpoints: there are no unnecessary elements involved in a call.

We note that there are many services that involve the use of servers or proxy endpoints that communicate directly with clients. Since these endpoints are directly involved in providing service, it is necessary and appropriate for them to maintain state. Examples of proxy endpoints include application layer firewalls, caching servers, transcoders, network-based conference bridges, interactive voice response systems, and PSTN gateways. The DCS architecture models these as end-points, that maintain appropriate call state.

We now turn to the mechanisms that allow us to avoid state in the DCS-proxies. A number of examples of the need for distributed state arise in the implementation of telephony features. These give rise to two types of information that a DCS-proxy may present to an endpoint that may subsequently be given back to the proxy by the endpoint. The first type of information is Remote endpoint identification, contained in the "Remote-Party-ID" header. The second type of information is associated with an active session that an endpoint is participating in. This latter information, stored in the "State" header, is information that a service provider or proxy may need for methods that are invoked by an endpoint related to that session. Thus, a DCS-proxy stores the state information about the calls at an endpoint in two new headers, "State" and "Remote-Party-ID". The State header is both encrypted and signed by the proxy to ensure the privacy and the integrity of the information contained in the header. The information that may be contained in State includes resource information (such as Gate information) and billing information (such as a billing id). The Remote-Party-ID is only encrypted when privacy is requested by the endpoint (covered in detail in the Section 7 below.)

When needed, the endpoint provides the State to the DCS-proxy that generated it, which can use the information to provide additional functionality.
Because the State header is encrypted and signed by the DCS-proxy, the information it contains is trusted by the network even though the endpoint itself is not trusted. In addition, DCS-proxies store

service-specific opaque data associated with a call at the edge

router. Since charging for telephony services may be tied to the use of resources, this information is best stored at the edge router, where knowledge of resource usage exists.

The endpoint returns the state (possibly both State and Remote-Party-ID) information to the DCS-proxy when it is needed to implement specific features. The endpoint cannot interpret the information in the encrypted and signed State header (and Remote-Party-ID if it is also encrypted), and any attempt to tamper with it can be detected by the DCS-proxy.

An example of use of the State information is one where a change in coding method in the middle of a call (e.g., upon detection of a fax tone) may require the proxies to authorize additional resources. Services such as call-transfer and three-way-calling require the proxy to be involved in authorizing resources for packet flows to the new destination(s).

7. DCS Proxy - DCS Proxy Communications

DCS-proxies implement a set of service-specific control functions required to support the telephony service:

. Authentication and authorization: Since services are only provided to authorized subscribers, DCS-proxies authenticate signaling messages and authorize requests for service on a call-by-call basis.

. Name/number translation and call routing: DCS-proxies translate dialed E.164 numbers, or names, to a terminating IP address based on call routing logic to support a wide range of call features.

. Service-specific admission control: DCS-proxies can implement a broad range of admission control policies for the telephony service. For example, DCS-proxies may provide precedence for particular calls (e.g., 911 calls). Admission control may also be used to implement overload control mechanisms, e.g. to restrict the number of calls to a particular location or to restrict the frequency of call setup to avoid signaling overload.

. Signaling and service feature support: While many service features are implemented by endpoints, the DCS-proxy also plays a role in feature support. DCS signaling provides a set of service primitives to end-points that are mediated by the DCS-proxy. The DCS-proxy is involved in implementing service features that depend on the privacy of calling information, e.g., caller-ID blocking. It also plays a role in supporting service features that require users to receive a consistent view of feature operation even when an endpoint is down. For example, while an endpoint may normally participate in call forwarding, the DCS-proxy can control call forwarding on behalf of an endpoint when the endpoint is down.

End-points MTA-o and MTA-t communicate through the DCS-Proxies DP-o and DP-t, as shown in Figure 2. The interface of concern in this section is the one between the DCS-Proxies DP-o and DP-t. In contrast to a true stateless SIP proxy, the DCS-Proxy maintains transaction state. During the interval that a call is being setup, a DCS-Proxy keeps state related to a request until a response is received.

For each call made to a phone number, DP-o may need to perform the functions needed for Local Number Portability (LNP). If a LNP database lookup is performed and the resulting dialed string is modified, DP-o must modify the Request-URI to include the result of the LNP lookup. The originating proxy DP-o generates and stores the State header. This information is intended to be sent to endpoint MTA-o and included with the first response that is returned to MTA-o. The originating DCS-Proxy, DP-o, may then use the call state information provided to it in the State header to manipulate call-legs when requested by MTA-o.

As with conventional SIP proxies, DP-o adds its address to the top of the Via: header list with a branch=1 field when forwarding the request. In addition, to support billing functions for a carrier, DP-o appends opaque information called the Billing-Info and Billing-ID. In addition, to support the resource management functions (such as manipulating Gates for resource management in concert with call-leg manipulation), a Gate-Location: header is included. This allows for the subsequent generation of requests for access network QoS by the end-points.

We also depend on originating DCS-Proxy, DP-o to be responsible for manipulating call legs. For instance, when a call is being forwarded, information about the new destination that the call is being forwarded to is provided by DP-t to DP-o. The new INVITE is then issued from DP-o. The information exchanged between the DCS-proxies enables such a function to be performed.

8. Privacy

Many conventional telephony systems have the ability to provide information about the identity of the calling party to the called party before the latter accepts the call (such a capability is typically termed "Caller-ID"). Systems that support Caller-ID usually provide a mechanism that allows the calling party to instruct the network to refrain from delivering this information to the destination.

In order for an IP-based network to provide a caller with a similar capability, a new SIP header is needed to signal the desire for anonymity to the network elements that would otherwise provide the caller's identity to the destination party. If a caller desires to remain anonymous, several additional changes to standard SIP are necessary.

The triplet {From:, To:, Call-ID:} is used to identify a call leg in both endpoints and in proxies. Because call state information is pushed to the edge of the network, this information must be delivered unchanged to the destination endpoint.

The SIP From: header normally contains information that identifies the caller. In order to hide the identity of the caller, the From: header information is encrypted with the originating endpoint's key. The destination endpoint does not possess the key to decrypt the From: information. No new syntax for SIP is introduced here.

Normally, the SIP Call-ID: header also contains information about the caller. In the DCS architecture, to support privacy the value of the Call-ID: header is a cryptographic hash string that contains no information about the user.

Since all the normally available mechanisms for passing information about the caller are no longer available, a new SIP header, Remote-Party-ID, is used to pass the caller's identity to the destination. The Remote-Party-ID header is primarily used for endpoint identification. This header contains the information that would normally be present in the From: header; the network passes it to the destination endpoint only if the caller has not requested anonymity. If the caller had requested anonymity, then the Remote-Party-ID header contains an encrypted string that can be used by the proxy in handling further requests.

If the user at an endpoint wants to return the last call (e.g., by dialing *69 on a traditional telephone) the "call return" function is invoked. If the user had subscribed to the caller ID service feature, the terminating endpoint could store the information (phone number or IP address) associated with the last call. However, it may be the case that the user does not subscribe to the feature, or the originator of the previous call may have requested that this information be blocked in order to retain privacy. In this case, call return can be implemented, while keeping the caller's identity private, by using the encrypted Remote-Party-ID header.

In addition to the usual privacy elements provided by telephone systems, IP-based systems must implement methods of hiding the source IP address from the destination if the caller requires privacy. The entire address must be obscured, since even a few address bits may provide partial location information. Likewise, IP addresses of the destination should not be revealed to the caller, in order to maintain privacy of transfer destinations.

IP addresses typically appear in the Contact: header; they also appear in SDP descriptions contained in SIP messages. These must all be protected. We chose to use an application-level anonymizer that inspects the SIP call signaling messages and replaces any identifying information contained therein in a consistent manner. The identifying information is modified such that when the messages

are delivered to the destination endpoint any identifying information has been replaced with fields that obscure the identity of the party seeking privacy.

This mechanism does not require any modification to the call signaling initiated by the endpoints: the application-level anonymizer performs these functions silently within the network.

9. Security Considerations

Detailed security considerations related to this architecture will be addressed in a future companion draft.

10. Call Flows

This section contains sample message flows between MTAs, DCS-Proxies, and Call Management Systems (CMSs, which are trusted User Agents, as described in section 3.4).

The first four subsections provide details for handling of basic calls, between all combinations of MTAs and CMSs.

Following subsections provide details of the handling of call features, such as call-forwarding, call-transfer, three-way-calling, CODEC changes, operator services, electronic surveillance, and IP address privacy.

10.1 Basic Call Flow from MTA to MTA

```
   MTA-o      ER-o     Proxy-o     Proxy-t     ER-t      MTA-t

    | (1)Invite           |          |          |          |
    |----------|--------->|(2)Invite |          |          |
    |          |          |--------->|          |          |
    |          |          |          |(3)Invite |          |
    |          |          |          |----------|--------->|
    |          |          |          |          |(4)183 SDP|
    |          |          |          |<---------|----------|
    |          |          |          |          |          |
    |          |  Gate    |(5)183 SDP|Gate Setup|          |
    |          |  Setup   |<---------|--------->|          |
    |          |<---------|          |          |          |
    |  (6)183 SDP         |          |          |          |
```

```
|<---------|----------|          |          |          |
|          |          |(7)PRACK  |          |          |
|----------|----------|----------|----------|--------->|
|          |(8)200 OK (acknowledging PRACK) |          |
|<---------|----------|----------|----------|----------|
|          |          |          |          |          |
|<---------|--------Reserve Resources-------|--------->|
|          |          |          |          |          |
|          |           (9)COMET             |          |
|----------|--------- |----------|----------|--------->|
|             (10)200 OK (acknowledging COMET)         |
|<---------|----------|----------|----------|----------|
|          |          |          |        (11)180 Ring |
|          |          | (12)180  |<---------|----------|
|        (13)180 Ring |<---------|          |          |
|<---------|----------|          |          |          |
|          |          |(14)PRACK |          |          |
|----------|----------|----------|----------|--------->|
|          | (15)200 OK (acknowledging PRACK)          |
|<---------|----------|----------|----------|----------|
|          |          |          |          |          | User
|          |          |          |         (16)200 OK  | Answers
|          |          |  (17)200 |<---------|----------|
|         (18)200 OK  |<---------|          |          |
|<---------|----------|          |          |          |
|    (19)ACK          |          |          |          |
|----------|----------|----------|----------|--------->|
|          |          |          |          |          |
|<---------|----------Active  Call----------|--------->|
|          |          |          |          |          | User
|          |          |          |          (20)BYE    | Hangs Up
|<---------|----------|----------|----------|----------|
|          |          |          |         (21)200 OK  |
|----------|----------|----------|----------|--------->|
```


The above figure shows the basic DCS call flow from one MTA to another. The basic DCS call flow starts with an INVITE from MTA-o to MTA-t through proxies Proxy-o and Proxy-t. It follows the conventions of SIP. The Via headers are used to track the path of the request (INVITE) so that the response can traverse backwards through the same path.

This INVITE is sent requesting that MTA-t not ring until the QoS preconditions are met. The purpose of this first INVITE is to invoke call features, such as call forwarding, to determine the proper destination MTA, and to negotiate the bandwidth and codec to be used so that the proper resources can be reserved. The response (183-Session-Progress) acknowledges the receipt of the INVITE message, provides the SDP for the forward media flow, and provides contact information for end-to-end messages that happen later in the call flow. When the INVITE is received, MTA-t's state reflects that a call is now being set-up. After MTA-o receives the 183-Session-Progress, it sends a PRACK message directly to MTA-t (as specified in the contact header) to acknowledgement receipt of MTA-t's SDP.

After resources are reserved for the call, a COMET is sent to MTA-t. MTA-t responds with a 200-OK, and also sends a ringback indication in the form of a 180-Ringing message. When the call is answered, a 200-OK to the INVITE is sent back to MTA-o, which ACKs the OK to MTA-t to complete the triple handshake.

The bearer channel session can now be established. When the call is over, either end can send a BYE message directly to the other end. This BYE request must also be responded to with a 200-OK.

The detailed steps followed and messages exchanged are:

A call setup begins when MTA-o detects off-hook on one of its lines. MTA-o first puts that line in the "busy" state. MTA-o sends an audible dialtone signal to the customer and begins to detect DTMF digits. Upon receiving the first digit, MTA-o stops dialtone. Once a complete E.164 number has been received (based upon a digit map that has been provisioned in the MTA), MTA-o generates the following INVITE message and sends it to Proxy-o (the DCS-proxy that manages MTA-o). MTA-o starts the retransmission timer (T-proxy-request).

(1) INVITE

```
INVITE sip:555-2222@Host(DP-o);user=phone SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
Supported: 100rel, state
Remote-Party-ID: John Doe <tel:555-1111>
Anonymity: Off
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
         seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
```



```
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(mta-o.provider)
```

```
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuite:312F
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

The Request-URI starts with the dialed number from the user. The Remote-Party-ID gives the calling name and number, as provided by the MTA. Even though Anonymity indicates calling name and number privacy is not required for this call, the From, To, and Call-ID headers contain cryptographically random values to maintain privacy of the caller.

Upon receiving the INVITE message, Proxy-o authenticates MTA-o using standard IPSec authentication. Proxy-o examines the Remote-Party-ID: line and checks to see that this originating phone number belongs to MTA-o, and is authorized for originating service. Proxy-o also checks to make sure the calling name in the Remote-Party-ID: line is a valid calling name for this line. Proxy-o then sends the dialed number to a directory server for resolution to an IP address. In this example, the directory server returns the address of Proxy-t, the Proxy that manages the terminating MTA. Proxy-o generates the following INVITE message and sends it to Proxy-t. Proxy-oadds a number of parameters to the INVITE message, which are described below. Upon sending this INVITE message, Proxy-o starts the retransmisison timer (T-proxy-request) and starts the T3 session timer (T-proxy-setup). The retransmission timer is cancelled on receipt of the optional 100-Trying provisional response (not present in this call flow); both are cancelled on receipt of the 183-Session-Progress provisional response.

(2) INVITE

```
INVITE sip:+1-212-555-2222;rn=+1-212-234-2222;
        npdi=yes@Host(dp-t);user=phone  SIP/2.0
Via: SIP/2.0/UDP Host(DP-o.provider);branch=1
Via: SIP/2.0/UDP Host(mta-o.provider)
Supported: 100rel, state
Require: state
Proxy-Require: dcs, state
```

```
Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
Anonymity: Off
Dcs-Gate: Host(cmts-o.provider):3612/17S30124/37FA1948 required
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/
        212-555-1111/212-555-2222>
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=0
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

The "lrn" in the Request-URI shows that LNP dip has been done, and gives the result. The dialed number is fully expanded into E.164 number. The Remote-Party-ID header contains verified Calling-Name and full E.164 Calling-Number. Dcs-Gate contains the IP address of the CMTS, the identity of the originating gate, and key for gate coordination messages. Also contained is the indication that gate coordination is required for this call. Dcs-Billing-Info contains the IP address of the record-keeping-server for event collection, the account number, originating number, and terminating number for billing of this call. State contains the state information wanted by Proxy-o for handling of messages from MTA-t to MTA-o. Dcs-Billing-ID contains the unique Billing identifier, made up of the CMS IP address, timestamp, and sequence number. A suggested encryption key is inserted into the SDP.

Upon receiving this INVITE message, Proxy-t authenticates that the sender was Proxy-o using IPSec, and sends the E.164-t address to the directory server. In this example, the Directory Server is able to translate E.164-t to the IP address of MTA-t (one of the MTAs managed by Proxy-t). Proxy-t then checks to see if MTA-t is authorized for receiving this call. Proxy-talso checks the account

information to determine if MTA-o is paying for the call or if MTA-t is expected to pay. Proxy-t generates the following INVITE message and sends it to MTA-t. The Remote-Party-ID line appears unchanged only if the destination MTA has subscribed to caller-id service; otherwise, or if the caller had specified privacy of the caller information, the Remote-Party-ID line would be altered. Note that the Via lines have been encrypted, maintaining the privacy of the caller. The line State has been added, and contains all the information needed by the Proxy for any subsequent call features that may be requested. This information is signed by Proxy-t and encrypted.

Upon sending this INVITE message, Proxy-t starts the retransmisison timer (T-proxy-request) and starts the T3 session timer (T-proxy-setup). The retransmission timer is cancelled on receipt of the optional 100-Trying provisional response (not present in this call flow); both are cancelled on receipt of the 183-Session-Progress provisional response.

```
(3) INVITE
     INVITE sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"; via=Host(mta-o.provider)}K
     Supported: 100rel, state
     Require: state
     Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
     Media-Authorization: 31S14621
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider); gate=Host(cmts-t.provider):4321/31S14621;
             state="Host(dp-o.provider); nexthop=sip:555-
             1111@Host(mta-o.provider); gate=Host(cmts-
             o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
             2222; num-redirects=0"}K"
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
```

```
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
```



```
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Local number portability information has been removed from the Request-URI, and the username is a string that is known to MTA-t. Via headers are encrypted to provide calling party privacy. Media-Authorization header contains the Gate-ID at the CMTS controlling the resources for MTA-t. State contains a string encrypted with a Proxy-t privately-held key, and contains nexthop routing information, CMTS address, Gate-ID, and all previous state headers from other proxies.

Upon receiving this INVITE, MTA-t authenticates that the message came from Proxy-t using IPSec. MTA-t checks the telephone line associated with the E.164-t (as found in the Request URI) to see if it is available. If it is available, MTA-t looks at the capability parameters in the Session Description Protocol (SDP) part of the message and determines which media channel parameters it can accommodate for this call. MTA-t stores the INVITE message, including the encrypted State parameters, for later use. MTA-t puts this line in the "busy" state (so any other call attempts are rejected until this call clears), generates the following 183-Session-Progress response, and sends it to Proxy-t. MTA-t starts the retransmission timer with value (T-proxy-response) and starts the session timer (T3) with value (T-resource).

MTA-t can, at its option, still accept further incoming calls and present them all to the customer. Such enhanced user interfaces for the MTA is beyond the scope of this specification. Note that MTA-t can't use the To: header field to determine the proper line, as it may be totally unrelated to the phone number at MTA-t.

(4)183-Session-Progress

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
Require: 100rel
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
Remote-Party-ID: John Smith <tel:555-2222>
Anonymity: off
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(mta-t.provider)
```



```
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

Remote-Party-ID contains the called name and number, as provided by MTA. Anonymity indicates called name and number privacy is not requested for this call. SDP contains MTA-t's bearer channel IP address, and negotiated voice encoding parameters.

Upon receiving the 183-Session-Progress message, Proxy-t forwards the following message to Proxy-o, restoring the Via headers, and adding Dcs-Gate information. At this point Proxy-t has completed all the call processing functions needed for this call, deletes its local state information, and handles all remaining messages as a stateless proxy. Proxy-t may include Dcs-Billing-Information if it wishes to override the billing information that came in the INVITE (e.g. collect or toll-free call).

(5) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
Via: SIP/2.0/UDP Host(mta-o.provider)
Require: 100rel, state
Proxy-Require: dcs, state
State: Host(dp-t.provider); nexthop=sip:555-2222@Host(mta-
        t.provider); gate=Host(cmts-t.provider):4321/31S14621;
        orig-dest=tel:+1-212-555-1111; num-redirects=0
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=0
Dcs-Gate: Host(cmts-t.provider):4321/31S14621/37FA1948
Remote-Party-ID: John Smith <tel:+1-212-555-2222>
Anonymity: off
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
```



```
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, Proxy-o forwards the following message to MTA-o. This message contains a State parameter giving all the information needed by the Proxy for later features. The State value is signed by Proxy-o and encrypted by Proxy-o's privately-held key. At this point Proxy-o has completed all the call processing functions needed for this call, deletes its local state information, and handles all remaining messages as a stateless proxy.

(6) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: Sip/2.0/UDP Host(mta-o.provider)
Require: 100rel, state
Media-Authorization: 17S30124
State: Host(dp-o.provider); state="{gate= Host(cmts-
        o.provider): 3612/17S30124, nexthop=sip:+1-212-555-
        2222;rn=+1-212-2342222;npdi=yes@Host(DP-t),
        state="Host(dp- t.provider);
        nexthop=sip:555-2222@Host(mta-t.provider);
        gate=Host(cmts-t.provider):4321/31S14621; orig-
        dest=tel:+1-212-555-1111; num-redirects=0"}K"
Remote-Party-ID: John Smith <tel:+1-212-555-2222>
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
```



```
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, MTA-o stops timer (T-proxy-request) and sends the following PRACK message directly to MTA-t using the IP address in the Contact header of the 183-Session-Progress message.

(7) PRACK:

```
PRACK sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 128 PRACK
Rack: 9021 127 INVITE
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
```

MTA-t acknowledges the PRACK with a 200-OK, and begins to reserve the resources necessary for the call.

(8) 200 OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 128 PRACK
```

After sending PRACK(7), MTA-o attempts to reserve network resources if necessary. If resource reservation is successful, MTA-o sends the following COMET message directly to MTA-t. MTA-o starts timer (T-direct-request).

(9) COMET:

```
COMET sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 COMET
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
a=qos:success send
```

MTA-t acknowledges the COMET message with a 200-OK.

(10) 200 OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 COMET
```

Upon receipt of the 200-OK(10), MTA-o stops timer (T-direct-request).

Upon receipt of the (7) PRACK message, MTA-t stops timer (T-proxy-response) and attempts to reserve network resources if necessary. Once MTA-t both receives the COMET message and has successfully reserved network resources, MTA-t begins to send ringing voltage to the designated line and sends the following 180 RINGING message through Proxy-t. MTA-t restarts the session timer (T3) with value (T-ringing).

(11) 180 RINGING:

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
```

```
                o.provider); branch=1"; via=Host(mta-
o.provider)}
K
        Require: 100rel
```



```
        State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
                o.provider); gate=Host(cmts-t.provider):4321/31S14621;
                state="Host(dp-o.provider); nexthop=sip:555-
                1111@Host(mta-o.provider); gate=Host(cmts-
                o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
                2222; num-redirects=0"}K"
        From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
                seq=72))@localhost>
        To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
        Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
        Contact: sip:Host(mta-t.provider)
        Cseq: 127 INVITE
        Rseq: 9022
```

Proxy-t decodes the Via: headers, and passes the 180-Ringing to Proxy-o. This operation is done as a SIP stateless proxy.

```
   (12) 180 RINGING:
        SIP/2.0 180 Ringing
        Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
        Via: SIP/2.0/UDP Host(mta-o.provider)
        Require: 100rel
        State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
                o.provider); gate=Host(cmts-o.provider):3612/17S30124
        From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
                seq=72))@localhost>
        To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
        Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
        Cseq: 127 INVITE
        Rseq: 9022
```

Proxy-o handles the message as a SIP stateless proxy, and passes the 180-Ringing to MTA-o.

```
   (13) 180 RINGING:
        SIP/2.0 180 Ringing
        Via: SIP/2.0/UDP Host(mta-o.provider)
        Require: 100rel
        From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
                seq=72))@localhost>
```

```
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9022
```

Upon receipt of the 180 RINGING message, MTA-o restarts the transaction timer (T3) with value (T-ringing). MTA-o acknowledges the provisional response with a PRACK, and plays audible ringback tone to the customer.

(14) PRACK:

```
PRACK sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Contact: sip:Host(mta-t.provider)
Cseq: 130 PRACK
Rseq: 9022 127 INVITE
```

MTA-t acknowledges the PRACK with a 200-OK, and stops timer (T-proxy-response).

(15) 200 OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 130 PRACK
```

Once MTA-t detects off-hook on the called line, it disconnects ringing voltage from the line and sends the final response through the proxies. MTA-t stops timer (T-ringing) and starts timer (T-proxy-response). If necessary, MTA-t may also commit to resources that have been reserved for this call. At this point, MTA-t begins to generate bearer channel packets of encoded voice and send them to MTA-o using the IP address and port number specified in the SDP part of the original INVITE message.

(16) 200-OK:

```
        SIP/2.0  200 OK
        Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
                o.provider); branch=1"; via=Host(mta-
o.provider)}
K
        State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
                o.provider); gate=Host(cmts-t.provider):4321/31S14621;
                state="Host(dp-o.provider); nexthop=sip:555-
                1111@Host(mta-o.provider); gate=Host(cmts-
                o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
                2222; num-redirects=0"}K"
        From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
                seq=72))@localhost>
        To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
        Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
        Cseq: 127 INVITE
```

Proxy-t handles the message as a SIP stateless proxy, and forwards it to Proxy-o.

(17) 200-OK:

```
        SIP/2.0  200 OK
        Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
        Via: SIP/2.0/UDP Host(mta-o.provider)
        State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
                o.provider); gate=Host(cmts-o.provider):3612/17S30124
        From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
                seq=72))@localhost>
        To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
        Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
        Cseq: 127 INVITE
```

Proxy-o handles the message as a SIP stateless proxy, and forwards it to MTA-o.

(18) 200-OK:

```
        SIP/2.0  200 OK
        Via: SIP/2.0/UDP Host(mta-o.provider)
        From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
                seq=72))@localhost>
        To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
        Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
        Cseq: 127 INVITE
```

Upon receipt of the 200-OK message, MTA-o stops timer (T-ringing) and stops playing audible ringback tone to the customer and begins to play the bearer channel stream that is received from MTA-t. MTA-o sends the following ACK message to MTA-t. If necessary, MTA-o may also commit to resources that have been reserved for this call. At this point, MTA-o begins to generate bearer channel packets of encoded voice and send them to MTA-t using the IP address and port number specified in the SDP part of the original 183-Session-Progress message (that was a response to the original INVITE).

(19) ACK:

```
ACK sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 ACK
```

Upon receipt of the ACK message, MTA-t stop timer (T-proxy-response).

When either MTA detects hangup, it sends out a BYE message to the other MTA. In this example, MTA-o detected that the customer hung up the phone. MTA-o puts that line in the "idle" state so new calls can be made or received. It sends the following BYE message directly to MTA-t. MTA-o may also need to release network resources

that have been used for the call. MTA-o starts timer (T-direct-request).

(20) BYE:

```
BYE sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

Upon receipt of the BYE message, MTA-t stops playing the bearer channel stream received from MTA-o and, if necessary, releases network resources that have been used for this call. MTA-tsends the following 200-OK message to MTA-o. MTA-t starts a 15-second timer (T-hangup) (Note: this is a local interface issue, and not part of this specification). If MTA-t does not detect hangup on the line before timer (T-hangup) expires, it plays "reorder" tone on the customer line. Once hangup is detected, MTA-t puts that line in the "idle" state so new calls can be made or received.

```
(21) 200-OK:
     SIP/2.0  200 OK
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 131 BYE
```

Upon receipt of 200-OK, MTA-o stops timer (T-direct-request).

10.2 Basic Call Flow from MTA to CMS

```
MTA-o             Proxy-o         CMS-t           Endpoint-t

  | (1)Invite           |          |                     |
  |-------------------->|(2)Invite |                     |
  |                     |--------->|                     |
  |                     |          | Private Signaling   |
  |                     |          |<------------------->|
  |                     |(3)183 SDP|                     |
  |                     |<---------|                     |
```

```
    |                     |          |                     |
    |  (4)183 SDP         |          |                     |
    |<--------------------|          |                     |
    |              (5)PRACK          |                     |
    |---------------------|--------->|                     |
    |                     |          | Private Signaling   |
    |                     |          |<------------------->|
    | (6)200 OK (acknowledging PRACK)|                     |
    |<--------------------|----------|                     |
    |                     |          |                     |
    |<------------------Reserve Resources----------------->|
    |                     |          |                     |
    |             (7)COMET           |                     |
    |-------------------- |--------->|                     |
    |(8)200 OK (acknowledging COMET)                       |
    |<--------------------|----------|                     |
    |                     |          | Private Signaling   |
    |                     |          |<------------------->|
    |                     | (9)180   |                     |
    |   (10)180 Ringing   |<---------|                     |
    |<--------------------|          |                     |
    |             (11)PRACK          |                     |
    |---------------------|--------->|                     |
    |(12)200 OK (acknowledging PRACK)                      |
    |<--------------------|----------|                     |
    |                     |          |                     | User
    |                     |          | Private Signaling   | Answers
    |                     |          |<------------------->|
    |                     |  (13)200 |                     |
    |         (14)200 OK  |<---------|                     |
    |<--------------------|          |                     |
    |    (15)ACK          |          |                     |
    |---------------------|--------->|                     |
    |                     |          |                     |
    |<--------------------Active  Call-------------------->|
    |                     |          |                     | User
    |                     |          | Private Signaling   | Hangs Up
    |                  (16)BYE       |<------------------->|
    |<--------------------|----------|                     |
    |                 (17)200 OK     |                     |
    |---------------------|--------->|                     |
```

This section describes the DCS call signaling flow for a basic call that terminate on the PSTN, or some other endpoint controlled by a CMS.

A call setup begins when MTA-o detects off-hook on one of its lines. MTA-o first puts that line in the "busy" state. MTA-o sends an audible dialtone signal to the customer and begins to detect DTMF digits. Upon receiving the first digit, MTA-o stops dialtone. Once a complete E.164 number has been received (based upon a digit map that has been provisioned in the MTA), MTA-o generates the following SIP INVITE message and sends it to Proxy-o (the Proxy that manages MTA-o). MTA-o starts the retransmission timer (T-proxy-request).

(1) INVITE:

```
INVITE sip:555-2222@Host(DP-o);user=phone SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
Supported: 100rel, state
Remote-Party-ID: John Doe <tel:555-1111>
Anonymity: Off
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuite:312F
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving the INVITE message, Proxy-o authenticates MTA-o using standard IPSec authentication. Proxy-o examines the Remote-Party-ID: line and checks to see that this originating phone number belongs to MTA-o, and is authorized for originating service. Proxy-o also checks to make sure the calling name in the Remote-Party-ID: line is a valid calling name for this line. Proxy-o then sends the dialed number to a directory server for resolution to an IP address. In this example, the directory server returns the address of CMS-t, the

CMS that manages the endpoint device. Proxy-o generates the following INVITE message and sends it to CMS-t. Proxy-oadds a number of parameters to the INVITE message, which are described below. Upon sending this INVITE message, Proxy-o starts the retransmisison timer (T-proxy-request) and starts the T3 session timer (T-proxy-setup). The retransmission timer is cancelled on receipt of the optional 100-Trying provisional response (not present in this call flow); both are cancelled on receipt of the 183-Session-Progress provisional response.

(2) INVITE:

```
INVITE sip:+1-212-555-2222;rn=+1-212-234-2222;
        npdi=yes@Host(cms-t);user=phone SIP/2.0
Via: SIP/2.0/UDP Host(DP-o.provider);branch=1
Via: SIP/2.0/UDP Host(mta-o.provider)
Supported: 100rel, state
Require: state
Proxy-Require: dcs, state
Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
Anonymity: Off
Dcs-Gate: Host(cmts-o.provider):3612/17S30124/37FA1948 required
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
        555-1111/212-555-2222>
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=0
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
```

```
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE message, CMS-t authenticates that the sender was Proxy-o using IPSec, and determines the proper endpoint device to receive this call. CMS-t engages in local signaling with

that endpoint device, outside the scope of this specification, and determines the proper SDP for the media flow to this endpoint device. When complete, CMS-t forwards the following message message to Proxy-o. The CMS-t lists itself as the location of the Dcs-Gate, since it simulates the gate operation. CMS-t may include Dcs-Billing-Information if it wishes to override the billing information that came in the INVITE (e.g. collect or toll-free call).

(3) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
Via: SIP/2.0/UDP Host(mta-o.provider)
Require: 100rel, state
Proxy-Require: dcs, state
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=0
Dcs-Gate: Host(cms-t.provider):4321/137S90721/805628
Remote-Party-ID: John Smith <tel:+1-212-555-2222>
Anonymity: off
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(cms-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mg02.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
```

```
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, Proxy-o forwards the following message to MTA-o. This message contains a State parameter giving all the information needed by the Proxy for later features. The State value is signed by Proxy-o and encrypted by Proxy-o's privately-held key. At this point Proxy-o has completed all the call processing functions needed for this call, deletes its local state information, and handles all remaining messages as a stateless proxy.

(4) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
```



```
Via: Sip/2.0/UDP Host(mta-o.provider)
Require: 100rel, state
Media-Authorization: 17S30124
State: Host(dp-o.provider); state="{gate= Host(cmts-
        o.provider): 3612/17S30124, nexthop=sip:+1-212-555-
        2222;rn=+1-212-234-2222;npdi=yes@Host(cms-t)}K"
Remote-Party-ID: John Smith <tel:+1-212-555-2222>
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(cms-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mg02.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a-X=pc-qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, MTA-o stops timer

(T-proxy-request) and sends the following PRACK message directly to CMS-t using the IP address in the Contact header of the 183-Session-Progress message.

(5) PRACK:

```
PRACK sip:Host(cms-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 128 PRACK
Rack: 9021 127 INVITE
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
```



```
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
a-Qos:mandatory sendrecv
```

CMS-t acknowledges the PRACK with a 200-OK, and performs local signaling with the endpoint (outside the scope of this specification) in order to begin reserving the resources necessary for the call.

(6) 200 OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 128 PRACK
```

After sending PRACK(5), MTA-o attempts to reserve network resources if necessary. If resource reservation is successful, MTA-o sends the following COMET message directly to CMS-t. MTA-o starts timer (T-direct-request).

(7) COMET:

```
COMET sip:Host(cms-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 COMET
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
a=qos:success send
```

CMS-t acknowledges the COMET message with a 200-OK.

(8) 200 OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(mta-o.provider)
```



```
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 COMET
```

Upon receipt of the 200-OK(8), MTA-o stops timer (T-direct-request). Upon receipt of the (5) PRACK message, CMS-t stops timer (T-proxy-response) and signals the endpoint device to attempt to reserve the network resources necessary. Once CMS-t both receives the COMET message and acknowledgement from the endpoint device, CMS-t sends the following 180-Ringing (or 183-Sesison-Progress, with a Session:Media header) message. CMS-t restarts the session timer(T3) with value (T-ringing).

(9) 180 RINGING:

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
Via: SIP/2.0/UDP Host(mta-o.provider)
Require: 100rel
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
         seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Contact: sip:Host(cms-t.provider)
Cseq: 127 INVITE
Rseq: 9022
```

Proxy-o handles the message as a SIP stateless proxy, and passes the 180-Ringing (or 183-Session-Progress) to MTA-o.

(10) 180 RINGING:

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP Host(mta-o.provider)
Require: 100rel
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Contact: sip:Host(cms-t.provider)
Cseq: 127 INVITE
RSeq: 9022
```

Upon receipt of the 180 RINGING message, MTA-o restarts the transaction timer (T3) with value (T-ringing). MTA-o acknowledges the provisional response with a PRACK, and plays audible ringback tone to the customer.

(11) PRACK:

```
PRACK sip:Host(cms-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
```



```
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 130 PRACK
RAck: 9022 127 INVITE
```

CMS-t acknowledges the PRACK with a 200-OK, and stops timer (T-proxy-response).

```
(12) 200 OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 130 PRACK
```

Once CMS-t, via private signaling with the endpoint device, detects off-hook on the called line, it sends the final response to the INVITE. CMS-t stops timer (T-ringing) and starts timer (T-proxy-response). If necessary, MTA-t may also commit to resources that have been reserved for this call. At this point, the endpoint device begins to generate bearer channel packets of encoded voice and send them to MTA-o using the IP address and port number specified in the SDP part of the original INVITE message.

```
(13) 200-OK:
     SIP/2.0  200 OK
     Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
     Via: SIP/2.0/UDP Host(mta-o.provider)
     State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
             o.provider); gate=Host(cmts-o.provider):3612/17S30124
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
```

Proxy-o handles the message as a SIP stateless proxy, and forwards it to MTA-o.

```
(14) 200-OK:
     SIP/2.0  200 OK
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
```


Upon receipt of the 200-OK message, MTA-o stops timer (T-ringing) and stops playing audible ringback tone to the customer and begins to play the bearer channel stream that is received. MTA-o sends the following ACK message to CMS-t. If necessary, MTA-o may also commit to resources that have been reserved for this call. At this point, MTA-o begins to generate bearer channel packets of encoded voice and send them to the remote endpoint using the IP address and port number specified in the SDP part of the original 183-Session-Progress message (that was a response to the original INVITE).

(15) ACK:

```
ACK sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 ACK
```

Upon receipt of the ACK message, CMS-t stop timer (T-proxy-response).

When MTA-o detects a hangup, or the endpoint device controlled by CMS-t detects a hangup, it sends out a BYE message to the other endpoint. In this example, CMS-t detected that the customer hung up the phone. CMS-t puts that line in the "idle" state so new calls can be made or received. It sends the following BYE message directly to MTA-o. CMS-t may also need to release network resources that have been used for the call. CMS-t starts timer (T-direct-request).

(16) BYE:

```
BYE sip:Host(mta-o.provider) SIP/2.0
Via: SIP/2.0/UDP Host(cms-t.provider)
From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

Upon receipt of the BYE message, MTA-o stops playing the bearer channel stream received from the endpoint device, and, if necessary, releases network resources that have been used for this call. MTA-o sends the following 200-OK message to CMS-t. MTA-o starts a 15-second timer (T-hangup) (Note: this is a local interface issue, and not part of this specification). If MTA-o does not detect hangup on the line before timer (T-hangup) expires, it plays "reorder" tone on the customer line. Once hangup is detected, MTA-o puts that line in the "idle" state so new calls can be made or received.

(17) 200-OK:

```
SIP/2.0  200 OK
Via: SIP/2.0/UDP Host(cms-t.provider)
```

```
      From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
```



```
      To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
              seq=72))@localhost>
      Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
      Cseq: 131 BYE
```

Upon receipt of 200-OK, CMS-t stops timer (T-direct-request).

10.3 Basic Call Flow from CMS to MTA

```
   Endpoint-o           CMS-o      Proxy-t                MTA-t

    | Private Signaling   |          |                     |
    |<------------------->|(1)INVITE |                     |
    |                     |--------->|   (2)INVITE         |
    |                     |          |-------------------->|
    |                     |          |   (3) 183 SDP       |
    |                     |          |<--------------------|
    |                     |(4)183 SDP|                     |
    |                     |<---------|                     |
    | Private Signaling   |          |                     |
    |<------------------->|          |   (5)PRACK          |
    |                     |------------------------------->|
    |                     | (6)200 OK (acknowledging PRACK)|
    |                     |<-------------------------------|
    |                     |          |                     |
    |<------------------Reserve Resources----------------->|
    |                     |          |                     |
    | Private Signaling   |          |                     |
    |<------------------->|             (7)COMET           |
    |                     |------------------------------->|
    |                     |(8)200 OK (acknowledging COMET) |
    |                     |<--------------------|----------|
    |                     |          | (9) 180 Ringing     |
    |                     |          |<------------------->|
    |                     | (10)180  |                     |
    | Private Signaling   |<---------|                     |
    |<------------------->|             (11)PRACK          |
    |                     |------------------------------->|
    |                     |(12)200 OK (acknowledging PRACK)|
    |                     |<-------------------------------|
    |                     |          |                     | User
    |                     |          | (13)200 OK          | Answers
    |                     |          |<--------------------|
```

```
    |                     |  (14)200 |                     |
    | Private Signaling   |<---------|                     |
    |<------------------->|    (15)ACK                     |
    |                     |------------------------------->|
    |                     |          |                     |
    |<--------------------Active  Call-------------------->|
    |                     |          |                     | User
    |                     |                  (16)BYE       | Hangs Up
    |                     |<-------------------------------|
```



```
    |                     |                 (17)200 OK     |
    |                     |------------------------------->|
```

This example shows a call originating on the PSTN and directed to a destination on the PacketCable network. We assume the same sequence of user behavior as in the basic call flow of section 10.1, only difference being the location of the originator.

A call setup begins when the endpoint device controlled by CMS-o detects an off-hook condition on one of its lines. This event is communicated to CMS-o through a private signaling exchange beyond the scope of this specification. CMS-o first puts that line in the "busy" state, and collects a complete E.164 number. As a result of a translation function performed by CMS-o, the destination is determined to be a DCS device served by Proxy-t. CMS-o generates the following SIP INVITE message and sends it to Proxy-t. CMS-o starts the retransmission timer (T-proxy-request) and starts the T3 session timer (T -setup). The retransmission timer is cancelled on receipt of the optional 100-Trying provisional response (not present in this call flow); both are cancelled on receipt of the 183-Session-Progress provisional response.

(1) INVITE:

```
     INVITE sip:+1-212-555-2222;rn=+1-212-234-2222;
             npdi=yes@Host(DP-t);user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(cms-o.provider);branch=1
     Supported: 100rel, state
     Require: state
     Proxy-Require: dcs, state
     Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
     Anonymity: Off
     Dcs-Gate: Host(cms-o.provider):3612/17S30124/37FA1948 optional
     Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
             555-1111/212-555-2222>
     Dcs-Billing-ID: Host(cms-o.provider):36123E5C:0152
     From: John Doe; <tel:+1-212-555-1111>
```

```
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(cms-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mg02.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
```



```
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE message, Proxy-t authenticates that the sender was CMS-o using IPSec, and sends the E.164-t address to the directory server. In this example, the Directory Server is able to translate E.164-t to the IP address of MTA-t (one of the MTAs managed by Proxy-t). Proxy-t then checks to see if MTA-t is authorized for receiving this call. Proxy-talso checks the account information to determine if MTA-o is paying for the call or if MTA-t is expected to pay. Proxy-t generates the following INVITE message and sends it to MTA-t. The Remote-Party-ID line appears unchanged only if the destination MTA has subscribed to caller-id service; otherwise, or if the caller had specified privacy of the caller information, the Remote-Party-ID line would be altered. Note that the Via lines have been encrypted, maintaining the privacy of the caller. The line State has been added, and contains all the information needed by the Proxy for any subsequent call features that may be requested. This information is signed by Proxy-t and encrypted.

Upon sending this INVITE message, Proxy-t starts the retransmisison timer (T-proxy-request) and starts the T3 session timer (T-proxy-setup). The retransmission timer is cancelled on receipt of the optional 100-Trying provisional response (not present in this call flow); both are cancelled on receipt of the 183-Session-Progress provisional response.

```
(2) INVITE:
     INVITE sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"}K
     Supported: 100rel, state
     Require: state
     Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
     Media-Authorization: 31S14621
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider);gate=Host(cmts-t.provider):4321/31S14621}K"
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
     Contact: sip:Host(cms-o.provider)
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mg02.provider)
     b=AS:64
     t=907165275 0
```



```
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     a=rtpmap:96 G726-32/8000
     m=audio 3456 RTP/AVP 0
     a=qos:mandatory sendrecv
     a=X-pc-codecs:96
```

Upon receiving this INVITE, MTA-t authenticates that the message came from Proxy-t using IPSec. MTA-t checks the telephone line associated with the E.164-t (as found in the Request URI) to see if it is available. If it is available, MTA-t looks at the capability parameters in the Session Description Protocol (SDP) part of the message and determines which media channel parameters it can accommodate for this call. MTA-t stores the INVITE message, including the encrypted State parameters, for later use. MTA-t puts this line in the "busy" state (so any other call attempts are rejected until this call clears), generates the following 183-Session-Progress response, and sends it to Proxy-t. MTA-t starts the retransmission timer with value (T-proxy-response) and starts

the session timer (T3) with value (T-resource).
MTA-t can, at its option, still accept further incoming calls and present them all to the customer. Such enhanced user interfaces for the MTA is beyond the scope of this specification. Note that MTA-t can't use the To: header field to determine the proper line, as it may be totally unrelated to the phone number at MTA-t.

(3) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider); branch=1"}K
Require: 100rel
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider);gate=Host(cmts-t.provider):4321/31S14621}K"
Remote-Party-ID: John Smith <tel:555-2222>
Anonymity: off
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
```



```
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, Proxy-t forwards the following message to CMS-o, restoring the Via headers, and adding Dcs-Gate information. At this point Proxy-t has completed all the call processing functions needed for this call, deletes its local state information, and handles all remaining messages as a stateless proxy. Proxy-t may include Dcs-Billing-Information if it wishes to override the billing information that came in the INVITE (e.g. collect or toll-free call).

(4) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
Require: 100rel, state
State: Host(dp-t.provider); nexthop=sip:Host(dp-o.provider);
        gate=Host(cmts-t.provider):4321/31S14621; orig-
        dest=tel:+1-212-555-1111; num-redirects=0
Dcs-Gate: Host(cmts-t.provider):4321/31S14621/37FA1948
Remote-Party-ID: John Smith <tel:+1-212-555-2222>
Anonymity: off
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, CMS-o stops timer (T-proxy-request) and sends the following PRACK message directly to MTA-t using the IP address in the Contact header of the 183-Session-Progress message.

(5) PRACK:

```
PRACK sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(cms-o.provider)
From: John Doe; <tel:+1-212-555-1111>
```



```
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 128 PRACK
Rack: 9021 127 INVITE
```

```
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mg02.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
a-qos:mandatory sendrecv
```

MTA-t acknowledges the PRACK with a 200-OK, and begins to reserve the resources necessary for the call.

(6) 200 OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(cms-o.provider)
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 128 PRACK
```

After sending PRACK(5), CMS-o signals to the endpoint device to attempt to reserve the network resources necessary for the connection. If the endpoint signals that resource reservation is successful, CMS-o sends the following COMET message directly to MTA-t. CMS-o starts timer (T-direct-request).

(7) COMET:

```
COMET sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(cms-o.provider)
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 COMET
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mg02.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
```

```
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     m=audio 3456 RTP/AVP 0
     a=qos:success send
```

MTA-t acknowledges the COMET message with a 200-OK.

(8) 200 OK:

```
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(cms-o.provider)
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 COMET
```

Upon receipt of the 200-OK(10), CMS-o stops timer (T-direct-request).

Upon receipt of the (5) PRACK message, MTA-t stops timer (T-proxy-response) and attempts to reserve network resources if necessary. Once MTA-t both receives the COMET message and has successfully reserved network resources, MTA-t begins to send ringing voltage to the designated line and sends the following 180 RINGING message through Proxy-t. MTA-t restarts the session timer(T3) with value (T-ringing).

(9) 180 RINGING:

```
     SIP/2.0 180 Ringing
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"}K
     Require: 100rel
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider);gate=Host(cmts-t.provider):4321/31S14621}K"
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Contact: sip:Host(mta-t.provider)
     Cseq: 127 INVITE
     Rseq: 9022
```

Proxy-t decodes the Via: headers, and passes the 180-Ringing to CMS-o. This operation is done as a SIP stateless proxy.

(10) 180 RINGING:

```
     SIP/2.0 180 Ringing
     Via: SIP/2.0/UDP Host(cms-o.provider);branch=1
     Require: 100rel
     From: John Doe; <tel:+1-212-555-1111>
```

```
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Contact: sip:Host(mta-t.provider)
     Cseq: 127 INVITE
     RSeq: 9022
```


Upon receipt of the 180 RINGING message, CMS-o restarts the transaction timer (T3) with value (T-ringing). CMS-o acknowledges the provisional response with a PRACK, and signals the endpoint device to play audible ringback tone to the customer.

```
(11) PRACK:
     PRACK sip:Host(mta-t.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(cms-o.provider)
     From: John Doe; &lt;tel:+1-212-555-1111&gt;
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 130 PRACK
     RAck: 9022 127 INVITE
```

MTA-t acknowledges the PRACK with a 200-OK, and stops timer (T-proxy-response).

```
(12) 200 OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(cms-o.provider)
     From: John Doe; &lt;tel:+1-212-555-1111&gt;
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 130 PRACK
```

Once MTA-t detects off-hook on the called line, it disconnects ringing voltage from the line and sends the final response through the proxies. MTA-t stops timer (T-ringing) and starts timer (T-proxy-response). If necessary, MTA-t may also commit to resources that have been reserved for this call. At this point, MTA-t begins to generate bearer channel packets of encoded voice and send them to MTA-o using the IP address and port number specified in the SDP part of the original INVITE message.

```
(13) 200-OK:
     SIP/2.0  200 OK
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"}K
     State: Host(dp-t.provider); state="{nexthop=sip:Host(cms-
             o.provider); gate=Host(cmts-t.provider):4321/31S14621;
```

```
         via="Host(cms-o.provider);branch=1"}K"
     From: John Doe; &lt;tel:+1-212-555-1111&gt;
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
```

Proxy-t handles the message as a SIP stateless proxy, and forwards it to CMS-o.

(14) 200-OK:

```
     SIP/2.0  200 OK
     Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
```



```
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
```

Upon receipt of the 200-OK message, CMS-o stops timer (T-ringing) and signals the endpoint device to stop playing audible ringback tone to the customer and to begin to play the bearer channel stream that is received from MTA-t. CMS-o sends the following ACK message to MTA-t. If necessary, CMS-o may also commit to resources that have been reserved for this call. At this point, the endpoint device begins to generate bearer channel packets of encoded voice and send them to MTA-t using the IP address and port number specified in the SDP part of the original 183-Session-Progress message (that was a response to the original INVITE).

(15) ACK:

```
     ACK sip:Host(mta-t.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(cms-o.provider)
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 ACK
```

Upon receipt of the ACK message, MTA-t stop timer (T-proxy-response).

When either endpoint detects hangup, it sends out a BYE message to the other one. In this example, MTA-t detected that the customer hung up the phone. MTA-t puts that line in the "idle" state so new calls can be made or received. It sends the following BYE message directly to CMS-o. MTA-t may also need to release network resources that have been used for the call. MTA-t starts timer (T-direct-request).

(16) BYE:

```
BYE sip:Host(cms-o.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-t.provider)
From: tel:+1-212-555-2222
To: John Doe; <tel:+1-212-555-1111>
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

Upon receipt of the BYE message, CMS-o signals the endpoint device to stop playing the bearer channel stream received from MTA-t and, if necessary, releases network resources that have been used for this call. CMS-o sends the following 200-OK message to MTA-t. Once hangup is detected on the endpoint device, CMS-o puts that line in the "idle" state so new calls can be made or received.

(17) 200-OK:

```
SIP/2.0  200 OK
```



```
Via: SIP/2.0/UDP Host(mta-t.provider)
From: tel:+1-212-555-2222
To: John Doe; <tel:+1-212-555-1111>
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

Upon receipt of 200-OK, MTA-t stops timer (T-direct-request).

10.4 Basic Call Flow from CMS to CMS

```
Endpoint-o           CMS-o       CMS-t           Endpoint-t

  | Private Signaling   |          |                     |
  |<------------------->|(1)INVITE |                     |
  |                     |--------->| Private Signaling   |
  |                     |          |<------------------->|
  |                     |(2)183 SDP|                     |
  |                     |<---------|                     |
  | Private Signaling   |          |                     |
  |<------------------->|(3)PRACK  |                     |
  |                     |--------->|                     |
  |                     |(4)200 OK | Private Signaling   |
  |                     |<-------->|<------------------->|
  |                     |          |                     |
  |<------------------Reserve Resources----------------->|
```

```
    |                     |          |                     |
    | Private Signaling   |          |                     |
    |<------------------->|(5)COMET  |                     |
    |                     |--------->|                     |
    |                     |(6)200 OK | Private Signaling   |
    |                     |<-------->|<------------------->|
    |                     | (7)180   |                     |
    | Private Signaling   |<---------|                     |
    |<------------------->|(8)PRACK  |                     |
    |                     |--------->|                     |
    |                     |(9)200 OK |                     |
    |                     |<---------|                     | User
    |                     |          | Private Signaling   | Answers
    |                     |          |<------------------->|
    |                     | (10)200  |                     |
    | Private Signaling   |<---------|                     |
    |<------------------->| (11)ACK  |                     |
    |                     |--------->|                     |
    |                     |          |                     |
    |<--------------------Active  Call-------------------->|
    |                     |          |                     | User
    |                     | (12)BYE  | Private Signaling   | Hangs Up
    |                     |<---------|<------------------->|
    |                     |(13)200 OK|                     |
    |                     |----------|                     |
```


A call setup begins when the endpoint device controlled by CMS-o detects an off-hook condition on one of its lines. This event is communicated to CMS-o through a private signaling exchange beyond the scope of this specification. CMS-o first puts that line in the "busy" state, and collects a complete E.164 number. As a result of a translation function performed by CMS-o, the destination is determined to be an endpoint device served by CMS-t. CMS-o generates the following SIP INVITE message and sends it to CMS-t. CMS-o starts the retransmission timer (T-proxy-request) and starts the T3 session timer (T-setup). The retransmission timer is cancelled on receipt of the optional 100-Trying provisional response (not present in this call flow); both are cancelled on receipt of the 183-Session-Progress provisional response.

(1) INVITE:

```
     INVITE sip:+1-212-555-2222;rn=+1-212-234-2222;
             npdi=yes@Host(cms-t);user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(cms-o.provider);branch=1
     Supported: 100rel, state
```

```
Require: state
Proxy-Require: dcs, state
Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
Anonymity: Off
Dcs-Gate: Host(cms-o.provider):3612/17S30124/37FA1948 optional
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
        555-1111/212-555-2222>
Dcs-Billing-ID: Host(cms-o.provider):36123E5C:0152
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(cms-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mg02.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE message, CMS-t authenticates that the sender was CMS-o using IPSec, and sends the E.164-t address to the directory server. In this example, the Directory Server is able to translate E.164-t to the IP address of one of the endpoint devices

controlled by CMS-t. CMS-t then checks to see if the endpoint device is authorized for receiving this call. CMS-talso checks the account information to determine if the originator is paying for the call or if the destination is expected to pay. CMS-t engages in private signaling exchange with the endpoint device, beyond the scope of this specification, and determines the SDP description of the media stream to be sent to this endpoint.

CMS-t puts this line in the "busy" state (so any other call attempts are rejected until this call clears), generates the following 183-Session-Progress response, and sends it to CMS-o. The Dcs-Gate header is omitted from this message, since CMS-o indicated it was optional, and CMS-t considers it optional as well. CMS-t starts the retransmission timer with value (T-proxy-response) and starts the session timer (T3) with value (T-resource). CMS-t may include Dcs-Billing-Information if it wishes to override the billing information that came in the INVITE (e.g. collect or toll-free call).

(2) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(cms-o.provider);branch=1
Require: 100rel, state
Proxy-Require: dcs, state
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(cms-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(rgw12.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, CMS-o stops timer (T-proxy-request) and sends the following PRACK message to CMS-t using the IP address in the Contact header of the 183-Session-Progress message.

(3) PRACK:

```
PRACK sip:Host(cms-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(cms-o.provider)
From: John Doe; <tel:+1-212-555-1111>
```



```
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 128 PRACK
```

```
     Rack: 9021 127 INVITE
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mg02.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     m=audio 3456 RTP/AVP 0
     a-qos:mandatory sendrecv
```

CMS-t acknowledges the PRACK with a 200-OK, and signals the endpoint device to begin to reserve the resources necessary for the call.

(4) 200 OK:

```
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(cms-o.provider)
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 128 PRACK
```

After sending PRACK(3), CMS-o signals to the endpoint device to attempt to reserve the network resources necessary for the connection. If the endpoint signals that resource reservation is successful, CMS-o sends the following COMET message to CMS-t. CMS-o starts timer (T-direct-request).

(5) COMET:

```
     COMET sip:Host(cms-t.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(cms-o.provider)
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 COMET
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mg02.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
```

```
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     m=audio 3456 RTP/AVP 0
     a=qos:success send
```

CMS-t acknowledges the COMET message with a 200-OK.

(6) 200 OK:

```
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(cms-o.provider)
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 COMET
```

Upon receipt of the 200-OK(6), CMS-o stops timer (T-direct-request). Upon receipt of the (3) PRACK message, CMS-t stops timer (T-proxy-response) and attempts to reserve network resources if necessary. Once CMS-t both receives the COMET message and has successfully reserved network resources, CMS-t signals the endpoint to begin to send ringing voltage to the designated line and sends the following 180 RINGING message. CMS-t restarts the session timer (T3) with value (T-ringing).

(7) 180 RINGING:

```
     SIP/2.0 180 Ringing
     Via: SIP/2.0/UDP Host(cms-o.provider);branch=1
     Require: 100rel
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Contact: sip:Host(cms-t.provider)
     Cseq: 127 INVITE
     Rseq: 9022
```

Upon receipt of the 180 RINGING message, CMS-o restarts the transaction timer (T3) with value (T-ringing). CMS-o acknowledges the provisional response with a PRACK, and signals the endpoint device to play audible ringback tone to the customer.

(8) PRACK:

```
     PRACK sip:Host(cms-t.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(cms-o.provider)
     From: John Doe; <tel:+1-212-555-1111>
     To: tel:+1-212-555-2222
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
```

```
Cseq: 130 PRACK
RAck: 9022 127 INVITE
```

CMS-t acknowledges the PRACK with a 200-OK, and stops timer (T-proxy-response).

(9) 200 OK:


```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(cms-o.provider)
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 130 PRACK
```

Once CMS-t detects off-hook on the called line, it disconnects ringing voltage from the line and sends the final response. CMS-t stops timer (T-ringing) and starts timer (T-proxy-response). If necessary, CMS-t may also commit to resources that have been reserved for this call. At this point, CMS-t signals to the endpoint device to begin to generate bearer channel packets of encoded voice and send them to the originating endpoint, at the IP address and port number specified in the SDP part of the original INVITE message.

(10) 200-OK:

```
SIP/2.0  200 OK
Via: SIP/2.0/UDP Host(cms-o.provider); branch=1
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
```

Upon receipt of the 200-OK message, CMS-o stops timer (T-ringing) and signals the endpoint device to stop playing audible ringback tone to the customer and to begin to play the bearer channel stream that is received from the destination endpoint. CMS-o sends the following ACK message to CMS-t. If necessary, CMS-o may also commit to resources that have been reserved for this call. At this point, the endpoint device begins to generate bearer channel packets of encoded voice and send them to the destination endpoint using the IP address and port number specified in the SDP part of the original 183-Session-Progress message (that was a response to the original INVITE).

(11) ACK:

```
ACK sip:Host(cms-t.provider) SIP/2.0
```

```
Via: SIP/2.0/UDP Host(cms-o.provider)
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 ACK
```

Upon receipt of the ACK message, CMS-t stop timer (T-proxy-response).

When either endpoint detects hangup, it sends out a BYE message to the other one. In this example, the originating endpoint detected that the customer hung up the phone. CMS-o puts that line in the "idle" state so new calls can be made or received. It sends the

following BYE message directly to CMS-t. CMS-o starts timer (T-direct-request).

(12) BYE:

```
BYE sip:Host(cms-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(cms-o.provider)
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

Upon receipt of the BYE message, CMS-t signals the endpoint device to stop playing the bearer channel stream received from the originator and, if necessary, releases network resources that have been used for this call. CMS-tsends the following 200-OK message to CMS-o. Once hangup is detected on the endpoint device, CMS-t puts that line in the "idle" state so new calls can be made or received.

(13) 200-OK:

```
SIP/2.0  200 OK
Via: SIP/2.0/UDP Host(cms-o.provider)
From: John Doe; <tel:+1-212-555-1111>
To: tel:+1-212-555-2222
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

Upon receipt of 200-OK, CMS-o stops timer (T-direct-request).

10.5 Call-Forwarding-Unconditional and Call-Forwarding-Busy

```
   MTA-o                Proxy-o      Proxy-t               MTA-t

     |      INVITE        |               |                   |
     |------------------->|    INVITE     |                   |
     |                    |-------------->|   (1)INVITE       |
     |                    |               |------------------>|
     |                    |               |  (2)302 Redirect  |
     |                    |               |<------------------|
     |                    |   (3)302      |                   |
     |                    |<--------------|   (4)ACK          |
     |                    |               |------------------>|
     |                    |   (5)ACK      |                   |
     |                    |-------------->|                   |
     |                    |
     |                    |             Proxy-f            MTA-f
     |                    |               |                   |
     |                    |  (6)INVITE    |                   |
     |                    |-------------->|   (7)INVITE       |
     |                    |               |------------------>|
     |                    |               |  (8)183 SDP       |
```

```
 |                  |               |<------------------|
 |                  |               |                   |
```

The initial call flow for Call-Forwarding-Unconditional/Busy is identical to that shown in Section 10.1 until MTA-t receives the following INVITE message from Proxy-t.

```
(1) INVITE:
     INVITE sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"; via=Host(mta-o.provider)}K
     Supported: 100rel, state
     Require: state
     Remote-Party-ID: John Doe <tel:+1-212-555-1111>
     Media-Authorization: 31S14621
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider); gate=Host(cmts-t.provider):4321/31S14621;
             state="Host(dp-o.provider); nexthop=sip:555-
             1111@Host(mta-o.provider); gate=Host(cmts-
             o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
             2222; num-redirects=0"}K"
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
```



```
     Contact: sip:Host(mta-o.provider)
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mta-o.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     a=rtpmap:96 G726-32/8000
     m=audio 3456 RTP/AVP 0
     a=qos:mandatory sendrecv
     a=X-pc-codecs:96
```

Upon receiving this message, MTA-t determines that the line associated with 212-555-2222 is having all calls forwarded. It may initiate some local action (e.g. to play special ringing tones) to provide notification that the call is being forwarded. It may perform some functions as a SIP proxy, using the received Call-ID and SDP description, to further locate the user. It then issues a REDIRECT (302) response to indicate that it wants the call forwarded. This message carries the forwarding number in the Contact header.

(2) 302-Redirect

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
Remote-Party-ID: John Smith <tel:555-2222>
Anonymity: off
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: tel:555-3333
```

Proxy-t verifies on receipt of the 302-Redirect message that the called party is a subscriber to the Call Forwarding service. Proxy-t also verifies that the called party is permitted to forward the call to the supplied destination. It then adds a DCS-billing field to the 302-Redirect message to allow the "second leg" of the forwarded call to be charged to the user associated with 212-555-2222. It also

restores the suppressed Via headers to allow the response to be routed back to Proxy-o.

(3) 302-Redirect

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP Host(dp-o.provider); branch = 1
Via: SIP/2.0/UDP Host(mta-o.provider)
Proxy-Require: dcs
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
        555-1111/212-555-2222>
```

```
Dcs-Billing-Info: Host(rks-t.provider)<4278-9865-8765-9000/212-
        555-2222/212-555-3333>
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=0
Remote-Party-ID: John Smith <tel:+1-212-555-2222>
Anonymity: off
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: tel:+1-212-555-3333
```

Proxy-t also sends an ACK to MTA-t.

(4) ACK

```
ACK sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 ACK
```

The transaction at MTA-t is now complete.

Proxy-o matches the 302 response to the INVITE it had sent out. It sends an ACK back to Proxy-t .

(5) ACK

```
ACK sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
Via: Sip/2.0/UDP Host(dp-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
CSeq: 127 ACK
```

The transaction at Proxy-t is now complete.

Proxy-o determines the Proxy-f for the E.164 number 212-555-3333 when it receives the 302-Redirect message. It generates an INVITE message and sends it to Proxy-f. It embeds two Dcs-Billing-Info headers in this message. The first one identifies the user associated with the E.164 number 212-555-1111 as paying for the initial call leg (212-555-1111/212-555-2222). The second one identifies the user associated with the E.164 number 212-555-2222 as paying for the second call leg (212-555-2222/212-555-3333). Proxy-o adds the Dcs-Redirect header giving the information about this call redirection.

(6) INVITE:

```
INVITE sip:+1-212-555-3333;rn=+1-212-265-3333;
        npdi=yes@Host(dp-f) ;user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-o.provider); branch = 2
Via: SIP/2.0/UDP Host(mta-o.provider);
Supported: 100rel, state
Require: state
Proxy-Require: dcs, state
Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
Anonymity: Off
Dcs-Gate: Host(cmts-o.provider):3612/17S30124/37FA1948 required
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
        555-1111/212-555-2222>
Dcs-Billing-Info: Host(rks-t.provider)<4278-9865-8765-9000/212-
        555-2222/212-555-3333>
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=1
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
CSeq: 127 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE, Proxy-f queries the directory server to determine the IP address (MTA-f) associated with 212-555-3333. It then forwards the INVITE message to MTA-f.

(7) INVITE:

```
INVITE sip:555-3333@Host(mta-f.provider); user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-f.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
Supported: 100rel, state
Require: state
Remote-Party-ID: John Doe <tel:+1-212-555-1111>
Media-Authorization: 22S21718
State: Host(dp-f.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-f.provider):4321/22S21718;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=1"}K"
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE, MTA-f authenticates that the message came from Proxy-f using IPSec. It checks the telephone line associated with the E.164-f to see if it is available. If it is available, MTA-f looks at the capability parameters in the Session Description Protocol (SDP) part of the message and determines which media channel parameters it can accommodate for this call. MTA-f stores the INVITE message, including the encrypted State parameters, for later use. MTA-f puts this line in the "busy" state (so any other call attempts are rejected until this call clears), generates the following 183-Session-Progress response, and sends it to Proxy-f. MTA-f starts timer (T-proxy-response).

(8) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(dp-f.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
Require: 100rel
State: Host(dp-f.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-f.provider):4321/22S21718;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=1"}K"
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(mta-f.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-f.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

The subsequent signaling call flows are identical to those shown in Section 10.1.

10.6 Call-Forwarding-No-Answer

```
   MTA-o               Proxy-o      Proxy-t               MTA-t

   ...                ...             ...                   ...
    |                  |               |                   |
    |                  |               |  (1)180 Ringing   |
    |                  |               |<------------------|
    |                  |(2)180 Ringing |                   |
    |  (3)180 Ringing  |<--------------|                   |
    |<-----------------|               |                   |
    |                  |  (4)PRACK     |                   |
    |----------------------------------------------------->|
    |                  |  (5)200 OK    |                   |
    |<-----------------------------------------------------|
    |                  |               |                   |
    |                  |               |                   |
    |                  |               |  (6)302 Redirect  |
    |                  |               |<------------------|
    |                  |   (7)302      |                   |
    |  (8)302 Redirect |<--------------|   (9)ACK          |
    |<-----------------|               |------------------>|
    |                  |  (10)ACK      |                   |
    |   (11) ACK       |-------------->|                   |
    |----------------->|                                   
    |                  |             Proxy-f              MTA-f
    |  (12) INVITE     |               |                   |
    |----------------->|  (13)INVITE   |                   |
```

```
 |                  |-------------->|  (14)INVITE       |
 |                  |               |------------------>|
 |                  |               |                   |
```

The Call Forwarding No Answer service is triggered when a called party does not pick up the phone after it rings for a pre-specified period of time. The subsequent call flow is different from that for the Call Forwarding Busy and Call Forwarding Unconditional services since the originating and terminating MTAs have already identified each other, have already reserved the resources for the call, and since the CMS/Proxies are no longer storing transaction state when the Forwarding function is triggered.

The initial set of messages for this service are the same as in Section 10.1 through the point at which MTA-t is ringing the phone, and MTA-o is generating ringback. For purposes of this example, consider the initial INVITE message received by MTA-t to be the following.

(not shown) INVITE:


```
INVITE sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
Supported: 100rel, state
Require: state
Remote-Party-ID: John Doe <tel:+1-212-555-1111>
Media-Authorization: 31S14621
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
```

```
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

In response to the INVITE message, MTA-t starts local ringback and sends a 180 RINGING notification to MTA-o. It also starts the timer (T-ringing).

(1) 180 RINGING:

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
Require: 100rel
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
```



```
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Contact: sip:Host(mta-o.provider)
Cseq: 127 INVITE
Rseq: 9022
```

The 180-Ringing message from Proxy-t to Proxy-o (2), the 180-Ringing message from Proxy-o to MTA-o (3), and the PRACK exchange (4) and (5), are identical to the basic call flow in Section 10.1, and not repeated here.
When the timer(T-ringing) at the MTA-t expires, it determines the forwarding number (555-3333) and sends a 302-Redirect response with this number in the Contact header.

(6) 302-Redirect

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
```

```
        o.provider); branch=1"; via=Host(mta-o.provider)}K
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: tel:555-3333
```

Proxy-t uses its IPSec association with MTA-t to determine the identity of the request. It then verifies the line subscribes to the Call-Forwarding-No-Answer service. Proxy-t uses its State value to recover the billing information for the current call (which is either stored directly in the State value, or stored indirectly with a pointer to the Gate which stores the billing information). Proxy-t adds an additional Dcs-Billing-Info header containing the billing information for the second leg of the forwarded call. Proxy-t converts the new destination number in the Contact header into a full E.164 number, and passes the 302-Redirect message to Proxy-o.

(7) 302-Redirect

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP Host(dp-o.provider); branch = 1
Via: SIP/2.0/UDP Host(mta-o.provider)
Proxy-Require: dcs
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
        555-1111/212-555-2222>
Dcs-Billing-Info: Host(rks-t.provider)<4278-9865-8765-9000/212-
        555-2222/212-555-3333>
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
```



```
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=0
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: tel:+1-212-555-3333
```

Proxy-o converts the Contact header into a private format URL containing the billing information and usage restrictions for the new call. By including a timestamp, Proxy-o insures the URL can't be used for later call attempts beyond those authorized by the forwarder. Also encoded in the URL is the information needed for the Dcs-Redirect header and any required LAES.

(8) 302-Redirect

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:{type=transfer; dest=tel:+1-212-555-3333; billing-
        info= Host(rks-o.provider)<5123-0123-4567-8900/212-555-
        1111/212-555-2222>; billing-info= Host(rks-
        t.provider)<4278-9865-8765-9000/212-555-2222/212-555-
        3333>; billing-id= Host(dp-o.provider):36123E5C:0152;
        expires=36123E9A; orig-dest=+1-212-555-2222;
        redirected-by=+1-212-555-2222; num-
        redirects=1}K@Host(dp-o.provider);user=private
```

Proxy-t sends the following ACK message to MTA-t after sending 302-Redirect(8).

(9) ACK

```
ACK sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 ACK
```

Proxy-o sends the following ACK message to Proxy-t after sending 302-Redirect(9).

(10) ACK

```
ACK sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-o.provider)
```



```
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
```

```
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 ACK
```

MTA-o sends the following ACK message to Proxy-o on receipt of the 302-Redirect(8).

```
(11) ACK
     ACK sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 ACK
```

The transaction at Proxy-o, Proxy-t and MTA-t is now complete.

MTA-o, if it so desires, may now initiate a new call to the destination given in the Contact header. To avoid confusion at MTA-o, the call leg identification for this new call is different from that of the previous call. Therefore, any stored State headers are not included in this INVITE, and only the Request-URI gives the handling and billing information.

```
(12) INVITE:
     INVITE sip:{type=transfer; dest=tel:+1-212-555-3333; billing-
             info= Host(rks-o.provider)<5123-0123-4567-8900/212-555-
             1111/212-555-2222>; billing-info= Host(rks-
             t.provider)<4278-9865-8765-9000/212-555-2222/212-555-
             3333>; billing-id= Host(dp-o.provider):36123E5C:0152;
             expires=36123E9A; orig-dest=+1-212-555-2222;
             redirected-by=+1-212-555-2222; num-
             redirects=1}K@Host(dp-o.provider);user=private SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Supported: 100rel, state
     Remote-Party-ID: John Doe <tel:555-1111>
     Anonymity: Off
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E98;
             seq=74))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E98; seq=75))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E98;seq=74))@localhost
     Cseq: 127 INVITE
     Contact: sip:Host(mta-o.provider)
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mta-o.provider)
```

```
     b=AS:64
     t=907165275 0
     a=X-pc-csuite:312F
     a=rtpmap:0 PCMU/8000
     a=rtpmap:96 G726-32/8000
     m=audio 3456 RTP/AVP 0
     a=qos:mandatory sendrecv
     a=X-pc- codecs:96
```

Proxy-o does all its normal authorization and authentication functions, and decodes the encrypted private username in the Request-URI. From that it builds the Dcs-Billing-Info, Dcs-Billing-ID, and Dcs-Redirect headers, and determines the destination address. The INVITE message sent on to Proxy-f is as follows.

```
(13) INVITE:
     INVITE sip:+1-212-555-3333;rn=+1-212-265-3333;
             npdi=yes@Host(dp-f);user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-o.provider);branch=2
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Supported: 100rel, state
     Require: state
     Proxy-Require: dcs, state
     Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
     Anonymity: Off
     Dcs-Gate: Host(cmts-o.provider):3612/3S73916/518C3B22 required
     Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
             555-1111/212-555-2222>
     Dcs-Billing-Info: Host(rks-t.provider)<4278-9865-8765-9000/212-
             555-2222/212-555-3333>
     State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
             o.provider); gate=Host(cmts-o.provider):3612/3S73916;
             orig-dest=tel:+1-212-555-2222; num-redirects=1
     Dcs-Billing-ID: Host(dp-o.provider):36123E98:0171
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E98;
             seq=74))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E98; seq=75))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E98;seq=74))@localhost
     Cseq: 127 INVITE
     Contact: sip:Host(mta-o.provider)
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
```

```
      s=-
      c= IN IP4 Host(mta-o.provider)
      b=AS:64
      t=907165275 0
      a=X-pc-csuites:312F
      a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
      a=rtpmap:0 PCMU/8000
      a=rtpmap:96 G726-32/8000
```



```
      m=audio 3456 RTP/AVP 0
      a=qos:mandatory sendrecv
      a=X-pc-codecs:96
```

The remainder of the call procedes as in Section 10.1.

10.7 Call-Forwarding-MTA-Unavailable

```
   MTA-o              Proxy-o      Proxy-t              MTA-t

     |                  |               |  (1) REGISTER     |
     |                  |               |<------------------|
     |                  |               |  (2)200 OK        |
     |                  |               |------------------>|
     |                  |               |                   |
     |   INVITE         |               |                   |
     |----------------->|   INVITE      |                   |
     |                  |-------------->|    INVITE         |
     |                  |               |------------------>|
     |                  |               |                   |
     |                  |           (Timeout)               |
     |                  |               |                   |
     |                  |  302 REDIRECT |                   |
     |                  |<--------------|                   |
     |                  |     ACK       |                   |
     |                  |-------------->|                   |
     |                  |
     |                  |             Proxy-f             MTA-f
     |                  |               |                   |
     |                  |    INVITE     |                   |
     |                  |-------------->|     INVITE        |
     |                  |               |------------------>|
     |                  |               |                   |
```

This service consists of two parts. First, the MTA must register the forwarding address with the Proxy. Later, when an incomming call is handled by the proxy and the MTA is not available, the Proxy initiates the call forwarding.

MTA-t recognizes that the customer dialed the code to activate Call Forwarding, and prompts the customer for the forwarding telephone number. This information is sent to the Proxy in a REGISTER message.

```
(1) REGISTER
     REGISTER sip:Host(dp-o.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-t.provider)
     From: sip:555-2222@Host(mta-t.provider); user=phone
     To: sip:Host(dp-o.provider)
     Call-ID: B64(SHA-1(555-2222;time=361013B8;seq=1))
     Cseq: 1 REGISTER
```



```
     Contact: tel:555-3333
     Expires: 7200
```

The Proxy validates that the forwarding number maps to either a MTA it knows about or to another valid Proxy. The Proxy also checks to make sure that the customer subscribes to the Call Forwarding service, and if so activates the service and stores the forwarding number for later use. It responds to the MTA with a 200-OK.

```
(2) 200-OK
     SIP 2.0 200 OK
     Via: SIP/2.0/UDP Host(mta-t.provider)
     From: sip:555-2222@Host(mta-t.provider); user=phone
     To: sip:Host(dp-o.provider)
     Call-ID: B64(SHA-1(555-2222;time=361013B8;seq=1))
     Cseq: 1 REGISTER
```

For an incomming call, the initial sequence of messages is identical to that for a regular call setup as shown in Section 10.1, until Proxy-t forwards the INVITE message to MTA-t. Proxy-t times out when it does not receive a response to the INVITE from MTA-t. It determines that the called party subscribes to Call Forwarding service and that the forwarding number is 212-555-3333. It then generates a SIP 302 (Redirect) message with the forwarding number in the Contact header. It then adds the DCS-billing and Dcs-Billing-ID fields to the 302 message that allows the second leg of the forwarded call to be charged to the user associated with 212-555-2222. The subsequent call flow is the same as with Call Forwarding Unconditional, (or Call Forwarding Busy), and is given in Section 10.5.

10.8 Return-Call

```
   MTA-o              Proxy-o      Proxy-t             MTA-t

     |  (1) INVITE      |               |                   |
     |----------------->|  (2) INVITE   |                   |
     |                  |-------------->|   (3) INVITE      |
     |                  |               |------------------>|
     |                  |               |  (4) 183 SDP      |
     |                  |  (5) 183 SDP  |<------------------|
     |   (6) 183 SDP    |<--------------|                   |
     |<-----------------|   (7) PRACK   |                   |
     |----------------------------------------------------->|
     |                  |   (8) 200 OK  |                   |
     |<-----------------------------------------------------|
     |                  |   (9) COMET   |                   |
     |----------------------------------------------------->|
     |                  |  (10) 200 OK  |                   |
     |<-----------------------------------------------------|
     |                  |               | (11) 180 Ringing  |
```

```
   |                  |(12)180Ringing |<------------------|
   | (13)180 Ringing  |<--------------|                   |
   |<-----------------|  (14) PRACK   |                   |
   |--------------------------------------------------------->|
   |                  |  (15) 200 OK  |                   |
   |<---------------------------------------------------------|
   |                  |               |                   |
   |  (16) CANCEL     |               |                   |
   |----------------->|  (17) CANCEL  |                   |
   |                  |-------------->|  (18) CANCEL      |
   |                  |               |------------------>|
   |                  |               |  (19) 200 OK      |
   |                  |  (20) 200 OK  |<------------------|
   |   (21) 200 OK    |<--------------|                   |
   |<-----------------|               |                   |
   |                  |               |                   |
   |                  |               |   (22) INVITE     |
   |                  | (23) INVITE   |<------------------|
   |                  |<--------------|                   |
```

We assume for this example that MTA-t had last received a call from MTA-o. The INVITE message forwarded to MTA-o included the Remote-Party-ID line, which contained, among other items, a URL that identified MTA-o. If the original caller did not request privacy, and the destination subscribed to caller-id, then the URL contains the E.164 number, which can be used to place the return call. We

assume for this example that was not the case, and that MTA-t does not know the identity of the new call's destination.

Messages (1) through (15) in the above diagram are identical to those for the basic call flow given in Section 10.1, and message (16) through (21) is a standard SIP CANCEL operation. The key parameters used in processing the return-call are contained in message (3), reproduced below. For purposes of this example, we assume the destination had not subscribed to Caller-ID service, and therefore the calling-name and calling-number information is not present in (3) INVITE.

(3) INVITE:

```
     INVITE sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"; via=Host(mta-o.provider)}K
     Supported: 100rel, state
     Require: state
```

```
Remote-Party-ID: <sip:{type=remote-id; orig=tel:+1-212-555-
        1111; otherstuff=whatever}K@Host(dp-t.provider);
        user=private>; rpi-id=na
Media-Authorization: 31S14621
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```


Upon the user dialing *69, MTA-t initiates a call by sending an INVITE message to its Proxy, with the Request-URI containing the URL for the call to be returned. The complete message is as follows.

```
(22) INVITE:
     INVITE sip:{type=remote-id; orig=tel:+1-212-555-1111;
             otherstuff=whatever}K@Host(dp-t.provider); user=private
             SIP/2.0
     Via: SIP/2.0/UDP Host(mta-t.provider)
     Supported: 100rel, state
```

```
       Remote-Party-ID: John Smith <tel:555-2222>
       Anonymity: Off
       From: sip:B64(SHA-1(555-2222; time=36123F12;seq=3))@localhost
       To: sip:B64(SHA-1(*69; time=36123F12;seq=4))@localhost
       Call-ID: B64(SHA-1(555-2222;time=36123F12;seq=3))@localhost
       Cseq: 127 INVITE
       Contact: sip:Host(mta-t.provider)
       Content-Type: application/sdp
       Content-length: (.)

       v=0
       o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
       s=-
       c= IN IP4 Host(mta-t.provider)
       b=AS:64
       t=907165275 0
       a=X-pc-csuites:312F
       a=rtpmap:0 PCMU/8000
       a=rtpmap:96 G726-32/8000
       m=audio 7242 RTP/AVP 0
       a=qos:mandatory sendrecv
       a=X-pc-codecs:96
```

Upon receiving the INVITE message, Proxy-t authenticates MTA-t using standard IPSec. Proxy-t decrypts the destination string using its privately-held key, and checks its signature in the result. From this string the real destination E.164 is extracted. Proxy-t checks the "Remote-Party-ID:" line, and checks to see that this line belongs to MTA-t, and has either subscribed to call-return service, or is authorized to use the service and be charged on a per-use basis. Proxy-t then performs all the regular call handling functions, as described in the basic call flow. The message sent to Proxy-o is the following, and the call proceeds identically to the basic call flow from this point onward.

```
   (23) INVITE:
       INVITE sip:+1-212-555-1111;rn=+1-212-237-1111;
               npdi=yes@Host(dp-o.provider);user=phone SIP/2.0
       Via: SIP/2.0/UDP Host(dp-t.provider);branch=1
       Via: SIP/2.0/UDP Host(mta-t.provider)
       Supported: 100rel, state
       Require: state
       Proxy-Require: dcs, state
```



```
       Remote-Party-ID: John Smith <tel:+1-212-555-2222>
       Anonymity: Off
```

```
Dcs-Gate: Host(cmts-t.provider):4321/31S14621/37FA1948
Dcs-Billing-Info: Host(rks-t.provider)<5098-0987-6543-2100/212-
        555-2222/212-555-1111/*69>
Dcs-Billing-ID: Host(dp-t.provider):36123F12:0381
State: Host(dp-t.provider); nexthop=sip:555-2222@Host(mta-
        t.provider); gate=Host(cmts-t.provider):4321/31S14621
From: sip:B64(SHA-1(555-2222; time=36123F12;seq=3))@localhost
To: sip:B64(SHA-1(*69; time=36123F12;seq=4))@localhost
Call-ID: B64(SHA-1(555-2222;time=36123F12;seq=3))@localhost
Cseq: 127 INVITE
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 7242 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Remainder of call proceeds identically to the basic call flow given in Section 10.1.

10.9 Customer-Originated-Trace

```
   MTA-o               Proxy-o      Proxy-t               MTA-t

    |  (1) INVITE      |               |                    |
    |----------------->|  (2) INVITE   |                    |
    |                  |-------------->|   (3) INVITE       |
    |                  |               |------------------->|
    |                  |               |  (4) 183 SDP       |
    |                  |  (5) 183 SDP  |<-------------------|
    |   (6) 183 SDP    |<--------------|                    |
    |<-----------------|   (7) PRACK   |                    |
    |------------------------------------------------------>|
    |                  |   (8) 200 OK  |                    |
    |<------------------------------------------------------|
    |                  |   (9) COMET   |                    |
    |------------------------------------------------------>|
    |                  |  (10) 200 OK  |                    |
    |<------------------------------------------------------|
    |                  |               | (11) 180 Ringing   |
    |                  |(12)180Ringing |<-------------------|
    | (13)180 Ringing  |<--------------|                    |
    |<-----------------|  (14) PRACK   |                    |
    |------------------------------------------------------>|
    |                  |  (15) 200 OK  |                    |
    |<------------------------------------------------------|
    |                  |               |                    |
    |                  |               |  (16) 200 OK       |
    |                  |  (17) 200 OK  |<-------------------|
    |   (18) 200 OK    |<--------------|                    |
    |<-----------------|               |                    |
    |                  |               |                    |
    |                  |               |   (19) INVITE      |
    |                  |    (20) INVITE|<-------------------|
    |                  |     <---------|                    |
```

Call-trace (*57) is almost identical to return-call (*69), but the action taken by the Proxy is to report the information to law enforcement authorities, and complete the call either to the Service

Provider's office or to an announcement server (which tells the customer to call the Service Provider's office).

(19) INVITE:

```
INVITE sip:call-trace@Host(dp-t.provider) SIP/2.0
```



```
Via: SIP/2.0/UDP Host(mta-t.provider)
Supported: 100rel, state
Dcs-Trace-Party-ID: sip:{type=remote-id; orig=tel:+1-212-555-
        1111; otherstuff=whatever}K@Host(dp-t.provider);
        user=private
Remote-Party-ID: John Smith <tel:555-2222>
Anonymity: Off
From: sip:B64(SHA-1(555-2222; time=36123F12;seq=3))@localhost
To: sip:B64(SHA-1(*57; time=36123F12;seq=4))@localhost
Call-ID: B64(SHA-1(555-2222;time=36123F12;seq=3))@localhost
Cseq: 127 INVITE
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 7242 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Proxy performs the reporting function and connects to either (1) an announcement server telling customer the information is recorded, and to now call the Business Office during normal business hours, or (2) the Business Office.

10.10 Call-Waiting

```
   MTA-o2     MTA-o1  Proxy-o      Proxy-t               MTA-t

     |  (1) INVITE      |                |                   |
     |----------------->|  (2) INVITE    |                   |
     |           |      |--------------->|   (3) INVITE      |
     |           |      |                |------------------>|
     |           |      |                |  (4) 183 SDP      |
     |           |      |  (5) 183 SDP   |<------------------|
     |   (6) 183 SDP    |<---------------|                   |
     |<-----------------|   (7) PRACK    |                   |
     |------------------------------------------------------>|
     |           |      |   (8) 200 OK   |                   |
     |<------------------------------------------------------|
     |           |      |   (9) COMET    |                   |
     |------------------------------------------------------>|
     |           |      |  (10) 200 OK   |                   |
     |<------------------------------------------------------|
     |           |      |                | (11) 180 Ringing  |
     |           |      |(12)180Ringing  |<------------------|
     | (13)180 Ringing  |<---------------|                   |
     |<-----------------|  (14) PRACK    |                   |
     |------------------------------------------------------>|
     |           |      |  (15) 200 OK   |                   |
     |<------------------------------------------------------|
     |           |      |        (16) INVITE(Hold)           |
```

```
|           |<-----------------------------------------|
|           |      |        (17) 200 OK                |
|           |----------------------------------------->|
|           |      |        (18) ACK                   |
|           |<-----------------------------------------|
|           |      |               |  (19) 200 OK      |
|           |      |  (20) 200 OK  |<------------------|
|   (21) 200 OK    |<--------------|                   |
|<-----------------|               |                   |
|           |      |               |                   |
|           |      |(22) INVITE(Hold)                  |
|<-----------------------------------------------------|
|           |      |  (23) 200 OK  |                   |
|----------------------------------------------------->|
|           |      |  (24) ACK     |                   |
|<-----------------------------------------------------|
|           |      |        (25) INVITE(Resume)        |
|           |<-----------------------------------------|
```



```
|           |      |        (26) 200 OK                |
|           |----------------------------------------->|
|           |      |        (27) ACK                   |
|           |<-----------------------------------------|
```

Call Waiting is a service that allows a customer to respond to an incoming call during the time the phone line is busy. The customer hears an audible alerting tone, and indicates acceptance of the new call via a hookflash (putting the previous call on hold). Subsequent hookflashes switch between the two active calls. The originator of the second call may hear a distinctive ringback tone. For this example, consider an existing call initiated by MTA-o1, with the following call identification:

```
MTA-t state for call from MTA-o1 to MTA-t
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Contact: sip:Host(mta-o1.provider)
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o1.provider); gate=Host(cmts-t.provider):4321/31S14621;
             state="Host(dp-o1.provider); nexthop=sip:555-
             1111@Host(mta-o1.provider); gate=Host(cmts-
             o1.provider):3612/17S30124; orig-dest=tel:+1-212-555-
```

```
          2222; num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-o1.provider)/04FA37<5123-0123-4567-
          8900/212-555-1111/212-555-2222>
     Dcs-Billing-ID: Host(dp-o1.provider):36123E5C:0152
```

The initial set of messages associated with the second arriving call, (1) through (13), as shown in the figure above, are very similar to those involved in a Basic Call Setup and are not explicitly enumerated below. After the initial INVITE exchange, the state information stored for this new call is:

```
MTA-t state for call from MTA-o2 to MTA-t
     From: sip:B64(SHA-1(555-3333; time=36124125;seq=23))@localhost
     To: sip:B64(SHA-1(555-2222; time=36124125;seq=24)@localhost
     Call-ID: B64(SHA-1(555-3333;time=36124125;seq=23))@localhost
     Contact: sip:Host(mta-o2.provider)
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
          o2.provider); gate=Host(cmts-t.provider):4321/32S35378;
          state="Host(dp-o2.provider); nexthop=sip:555-
          3333@Host(mta-o2.provider); gate=Host(cmts-
          o2.provider):3612/17S30124; orig-dest=tel:+1-212-555-
          2222; num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-o2.provider)/173F419B<6010-4500-
          6789-0123/212-555-3333/212-555-2222>
     Dcs-Billing-ID: Host(dp-o2.provider):36124125:0031
```


In response to the INVITE for the second incoming call, the user at MTA-t is provided some indication of the second call, e.g. using a special tone. If the user at MTA-t hits a flash hook in response to this, MTA-t issues a INVITE(Hold) message to MTA-o1 to put it on HOLD.

```
(16) INVITE (Hold):
     INVITE sip:Host(mta-o1.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-t.provider)
     From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
          seq=72))@localhost>
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 INVITE
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
```

```
s=-
c= IN IP4 0.0.0.0
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
```

MTA-o1 acknowledges the HOLD command with a 200-OK message. The response contains an updated SDP description for the stream to be received at MTA-o1, indicating an IP address of 0.0.0.0 for a held call.

(17) 200-OK

```
SIP/2.0 200  OK
Via: SIP/2.0/UDP Host(mta-t.provider)
From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 INVITE
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 0.0.0.0
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
```



```
m=audio 6522 RTP/AVP 0
```

MTA-t responds to the 200-OK message with the standard SIP ACK message. At this point it is safe for MTA-t to stop sending voice payload packets to MTA-o1 and not risk dropping the connection due to "dead MTA recovery."

(18) ACK

```
ACK Host(mta-o1.provider)
Via: SIP/2.0/UDP Host(mta-t.provider)
From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
```

```
     To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 ACK
```

Once the first conversation is successfully placed on hold, MTA-t indicates a completion to the "ringing" to MTA-o2.

```
(19) 200-OK
     SIP/2.0 200  OK
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"; via=Host(mta-o.provider)}K
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o2.provider); gate=Host(cmts-t.provider):4321/32S35378;
             state="Host(dp-o2.provider); nexthop=sip:555-
             3333@Host(mta-o2.provider); gate=Host(cmts-
             o2.provider):3612/17S30124; orig-dest=tel:+1-212-555-
             2222; num-redirects=0"}K"
     From: sip:B64(SHA-1(555-3333;time=36124125;seq=23))@localhost
     To: sip:B64(SHA-1(555-1111; time=36124125;seq=24)@localhost
     Call-ID: B64(SHA-1(555-3333;time=36124125;seq=23))@localhost
     Cseq: 128 INVITE
```

This 200-OK is passed through the proxy chain in messages (18) and (19) to MTA-o2. MTA-o2 responds with an acknowledgement, in a manor identical to the basic call flow.

```
(22) ACK
     ACK Host(mta-t.provider)
     Via: SIP/2.0/UDP Host(mta-o2.provider)
     From: sip:B64(SHA-1(555-3333;time=36124125;seq=23))@localhost
     To: sip:B64(SHA-1(555-1111; time=36124125;seq=24)@localhost
     Call-ID: B64(SHA-1(555-3333;time=36124125;seq=23))@localhost
     CSeq: 128 ACK
```

At this point the user at MTA-t has a connection to the second caller, MTA-o2, with the first caller, MTA-o1, on hold.

Subsequent hookflashes repeat the sequence of INVITE(hold)/200-OK/ACK to one destination, and INVITE(resume)/200-OK/ACK to the other. The INVITE (Hold) sequence (22) through (24) is identical to

(16) through (18). Once the 200-OK is received, it is safe for MTA-t to stop sending voice packets.

INVITE (Resume) is very similar, except that the SDP description includes the proper IP address in the "c=" line.

(25) INVITE (Resume):

```
INVITE sip:Host(mta-o1.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
CSeq: 130 INVITE
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
```

MTA-o1 acknowledges the Resume command with a 200-OK message. The response contains an updated SDP description for the stream to be received at MTA-o1, indicating the real IP address of Host(mta-o1.provider).

(26) 200-OK

```
SIP/2.0 200  OK
Via: SIP/2.0/UDP Host(mta-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
CSeq: 129 INVITE
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o1.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
```

```
     m=audio 3456 RTP/AVP 0
```

MTA-t responds to the 200-OK message with the standard SIP ACK message. At this point it is safe for MTA-t to start sending voice payload packets to MTA-o1.

```
(27) ACK
     ACK Host(mta-o.provider)
     Via: SIP/2.0/UDP Host(mta-t.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     CSeq: 129 ACK
```

10.11 Call-Transfer-Blind

```
   MTA-o              Proxy-o      Proxy-t             MTA-t

    |                  |               |                   |
    |<-----------------call-in-progress------------------->|
    |                  |               |  (1) REFER        |
    |                  |  (2) REFER    |<------------------|
    |   (3) REFER      |<--------------|                   |
    |<-----------------|               |                   |
    |                  |
    |                  |           Proxy-f             MTA-f
    |  (4) INVITE      |               |                   |
    |----------------->|  (5) INVITE   |                   |
    |                  |-------------->|   (6) INVITE      |
    |                  |               |------------------>|
    |                  |               |   (7) 183 SDP     |
    |                  |  (8) 183 SDP  |<------------------|
    |    (9) 183 SDP   |<--------------|                   :
    |<-----------------|               :                   :
    :                  :               :   (10) 200 OK     |
    :                  :  (11) 200 OK  |<------------------|
    |    (12) 200 OK   |<--------------|                   |
    |<-----------------|               |                   |
    |                  |
    |                  |           Proxy-t             MTA-t
    |  (13) 200 OK     |               |                   |
    |----------------->| (14) 200 OK   |                   |
    |                  |-------------->|  (15) 200 OK      |
    |                  |   (16) ACK    |------------------>|
    |<-----------------------------------------------------|
    |                  |   (17) BYE    |                   |
    |----------------------------------------------------->|
    |                  |  (18) 200 OK  |                   |
    |<-----------------------------------------------------|
```

```
|                  |
```

The Call Transfer service is triggered by the user by methods beyond the scope of this specification. Described in this section is a transfer service common known as "blind transfer" where the party initiating the transfer (MTA-t in this example) is not informed of the success or failure of the transfer operation. The alternative, commonly known as "consultative transfer" is described later.

For this example, consider an existing call initiated by MTA-o, with the following call identification:

```
MTA-t state for call from MTA-o to MTA-t
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Contact: sip:Host(mta-o.provider)
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider); gate=Host(cmts-t.provider):4321/31S14621;
             state="Host(dp-o.provider); nexthop=sip:555-
             1111@Host(mta-o.provider); gate=Host(cmts-
             o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
             2222; num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-o.provider)/04FA37<5123-0123-4567-
             8900/212-555-1111/212-555-2222>
     Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
```

When MTA-t desires to transfer the existing call, it determines the forwarding number (in this example 555-3333) and issues a REFER message to MTA-o. REFER is the same as a regular INVITE but includes an additional "Refer-to:" header and "Referred-by:" header. The "Refer-to:" header identifies the number to which the call needs to be forwarded, while the "Referred-by:" header identifies the existing call leg at MTA-o. The following message is sent to MTA-t's Proxy, Proxy-t.

```
(1) REFER:
     REFER sip: Host(mta-o.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-t.provider)
     Supported: 100rel, state
     Refer-to: tel:555-3333
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider); gate=Host(cmts-t.provider):4321/31S14621;
             state="Host(dp-o.provider); nexthop=sip:555-
```

```
            1111@Host(mta-o.provider); gate=Host(cmts-
            o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
            2222; num-redirects=0"}K"
    Remote-Party-ID: John Smith <tel:555-2222>
    Anonymity: off
    From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
    To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
            seq=72))@localhost>
    Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
    Cseq: 8001 INVITE
    Referred-by: sip:B64(SHA-1(555-2222; time=36123E5B;
            seq=73))@localhost
```

When the REFER is received at Proxy-t, it first verifies MTA-t has subscribed to Call Forwarding service. If so, it decrypts the State information to determine the local gate location and identification. Proxy-t queries the gate to obtain the call's billing information.

Proxy-t inserts billing information to indicate that the user associated with the number 212-555-2222 will pay for the new call segment. Proxy-t extracts the call routing from the Dcs-state information, and then forwards the message to Proxy-o.

(2) REFER:

```
    REFER sip: Host(dp-o.provider) SIP/2.0
    Via: SIP/2.0/UDP Host(dp-t.provider)
    Via: SIP/2.0/UDP Host(mta-t.provider)
    Supported: 100rel, state
    Proxy-Require: dcs
    State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
            o.provider); gate=Host(cmts-o.provider):3612/17S30124;
            orig-dest=tel:+1-212-555-2222; num-redirects=0
    Refer-to: tel:+1-212-555-3333? Dcs-Billing-Info= Host(rks-
            o.provider)<5123-0123-4567-8900/212-555-1111/212-555-
            2222> & Dcs-Billing-Info= Host(rks-t.provider)<4278-
            9865-8765-9000/212-555-2222/212-555-3333> & Dcs-
            Billing-ID= Host(dp-o.provider): 36123E5C:0152
    Remote-Party-ID: John Smith <tel:+1-212-555-2222>
    From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
    To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
            seq=72))@localhost>
    Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
    Cseq: 8001 INVITE
    Referred-by: sip:B64(SHA-1(555-2222; time=36123E5B;
            seq=73))@localhost
```

Proxy-o forwards the REFER message to MTA-o after encrypting the <Dcs-Billing-Info, Dcs-Billing-ID> headers.

```
(3) REFER:
     REFER sip: 555-1111@Host(mta-o.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(dp-o.provider), {via="Host(dp-
             t.provider); branch=1"; via=Host(mta-t.provider)}K
     Supported: 100rel, state
     Refer-to:  sip:{type=transfer; dest=tel:+1-212-555-3333;
             billing-id=Host(dp-o.provider): 36123E5C:0152;
             expires=<timestamp>; billing-info=Host(rks-
             o.provider)<5123-0123-4567-8900/212-555-1111/212-555-
             2222>; billing-info=Host(rks-t.provider)<4278-9865-
             8765-9000/212-555-2222/212-555-3333>; orig-dest=tel:+1-
             212-555-2222; redirected-by=tel:+1-212-555-2222; num-
             redirects=1}K@Host(dp-o.provider);user=private
     From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 8001 INVITE
     Referred-by: sip:B64(SHA-1(555-2222; time=36123E5B;
             seq=73))@localhost
```


After processing the REFER, MTA-o issues a INVITE to MTA-f. In addition to the standard headers carried in an INVITE message, the encrypted {Dcs-Billing-INFO, Dcs-Billing-ID} fields received in the REFER message are copied into the INVITE message. These fields indicate that the user associated with the 212-555-2222 number will be charged for the second call leg.

```
(4) INVITE:
     INVITE sip:{type=transfer; dest=tel:+1-212-555-3333; billing-
             id=Host(dp-o.provider): 36123E5C:0152;
             expires=<timestamp>; billing-info=Host(rks-
             o.provider)<5123-0123-4567-8900/212-555-1111/212-555-
             2222>; billing-info=Host(rks-t.provider)<4278-9865-
             8765-9000/212-555-2222/212-555-3333>; orig-dest=tel:+1-
             212-555-2222; redirected-by=tel:+1-212-555-2222; num-
             redirects=1}K@Host(dp-o.provider);user=private SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Supported: 100rel, state
     Remote-Party-ID: John Doe <tel:555-1111>
     Anonymity: Off
```

```
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E98;
        seq=74))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E98; seq=75))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E98;seq=74))@localhost
Cseq: 129 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

When the Proxy-o receives the INVITE it first decrypts the header information to find the real destination for the call. Proxy compares the current time against the timestamp in the encrypted string; if the request is too old, it is refused. It invokes the call routing logic to determine which Proxy (Proxy-f) to which the INVITE needs to be routed. It also embeds two Dcs-Billing-Info headers in this message. The first one identifies the user associated with the E.164 number 212-555-1111 as paying for the initial call leg (212-555-1111/212-555-2222). This information was derived from the customer account information for the caller during

the first call attempt. The second Dcs-Billing-Info header identifies the user associated with the E.164 number 212-555-2222 as paying for the second call leg (212-555-2222/212-555-3333), and was provided by Proxy-t in the REFER message.

(5) INVITE:

```
INVITE sip: +1-212-555-3333;rn=+1-212-265-3333;
        npdi=yes@Host(dp-f);user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-o.provider); branch=1;
Via: SIP/2.0/UDP Host(mta-o.provider);
Supported: 100rel, state
Require: state
```

```
Proxy-Require: dcs, state
Remote-Party-ID: John Doe <tel:+1-212-555-1111>
Anonymity: Off
Dcs-Gate: Host(cmts-o.provider):3612/17S30124/37FA1948
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
        555-1111/212-555-2222>
Dcs-Billing-Info: Host(rks-t.provider)<4278-9865-8765-9000/212-
        555-2222/212-555-3333>
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E98;
        seq=74))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E98; seq=75))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E98;seq=74))@localhost
Cseq: 129 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE, Proxy-f queries the directory server to determine the IP address (MTA-f) associated with 212-555-3333. It then forwards the INVITE message to MTA-f, after stripping off all of the billing fields, and adding the encrypted state information. This is identical to the basic call flow shown in Section 10.1, and is not repeated here.

Upon receipt of the 200-OK message, MTA-o sends the final response of the REFER, a 200-OK, to MTA-t. This message is routed through

the Proxy Proxy-o, Proxy-t, and then delivered to MTA-t. MTA-t responds directly with an ACK. Proxy-t is now done; MTA-o sends the BYE message, which follows immediately.

```
(13) 200-OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(dp-o.provider), {via="Host(dp-
             t.provider); branch=1"; via=Host(mta-t.provider)}K
     State: Host(dp-o.provider); state="{gate= Host(cmts-
             o.provider): 3612/17S30124, nexthop=sip:+1-212-555-
             2222;rn=+1-212-234-2222@Host(DP-t), state="Host(dp-
             t.provider); nexthop=sip:Host(dp-o.provider);
             gate=Host(cmts-t.provider):4321/31S14621; orig-
             dest=tel:+1-212-555-1111; num-redirects=0"}K"
     From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 8001 REFER
```

Proxy-o restores the encrypted Via headers, and forwards the OK to topmost Via - Proxy-t.

```
(14) 200-OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(dp-t.provider)
     Via: SIP/2.0/UDP Host(mta-t.provider)
     From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 8001 REFER
```

Proxy-tforwards the 200-OK to MTA-t.

```
(15) 200-OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(mta-t.provider)
     From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 8001 REFER
```

MTA-t responds with an ACK message.

```
(16) ACK:
     ACK sip:Host(mta-o.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-t.provider)
     From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
```

```
     Cseq: 8001 ACK



(17) BYE:
     BYE sip:Host(mta-t.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 BYE
```

Upon receipt of the BYE message, MTA-t releases all network resources that have been used for this call. MTA-t sends the following 200-OK message to MTA-o.

```
(18) 200-OK:
     SIP/2.0  200 OK
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 BYE
```

10.12 Call-Transfer-Consultative

Call Transfer with Consultation is triggered by the user by methods beyond the scope of this specification. It consists of two distinct phases: first placing the existing call on hold and placing a new call to another destination (the consultation), and secondly transfering the first call to the second destination (the transfer). For this example, consider an existing call initiated by MTA-t1 to MTA-o. The call identification information at MTA-o is as follows:

```
MTA-o state for call from MTA-t1 to MTA-o
     From: sip:B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
     To: tel:555-1111
     Call-ID: B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
     Contact: sip: Host(mta-t1.provider)
     Remote-Party-ID: tel:+1-212-555-2222
     State: Host(dp-o.provider); state="{nexthop=sip:Host(dp-
             t1.provider); gate=Host(cmts-o.provider):3612/17S30124;
             state="Host(dp-t1.provider); nexthop=sip:555-
             2222@Host(mta-t1.provider); gate=Host(cmts-
             t1.provider):4321/31S14621; orig-dest=tel:+1-212-555-
             1111; num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-t1.provider)<4278-9865-8765-
             9000/212-555-2222/212-555-1111>
     Dcs-Billing-ID: Host(dp-t1.provider):36124033:0381
```

MTA-o places this call on hold and determines the destination for consultation. MTA-o initiates a second call to the consultation endpoint, MTA-t2, as shown in the figure below.

```
   MTA-o              Proxy-o      Proxy-t2   MTA-t1     MTA-t2

     |  (1) INVITE(Hold)|               |        |          |
     |--------------------------------------->|          |
     |  (2) 200 OK      |               |        |          |
     |<---------------------------------------|          |
     |  (3) ACK         |               |        |          |
     |--------------------------------------->|          |
     |                  |               |        |          |
     |  (4) INVITE      |               |        |          |
     |----------------->|  (5) INVITE   |        |          |
     |                  |-------------->|   (6) INVITE      |
     |                  |               |------------------>|
     |                  |               |  (7) 183 SDP      |
     |                  |  (8) 183 SDP  |<------------------|
     |   (9) 183 SDP    |<--------------|        |          |
     |<-----------------|  (10) PRACK   |        |          |
     |----------------------------------------------------->|
     |                  |  (11) 200 OK  |                   |
     |<-----------------------------------------------------|
     |                  |  (12) COMET   |                   |
     |----------------------------------------------------->|
     |                  |  (13) 200 OK  |                   |
     |<-----------------------------------------------------|
     |                  |               | (14) 180 Ringing  |
     |                  |(15)180Ringing |<------------------|
     | (16)180 Ringing  |<--------------|                   |
     |<-----------------|  (17) PRACK   |                   |
```

```
 |---------------------------------------------------->|
 |                 |  (18) 200 OK  |                   |
 |<----------------------------------------------------|
 |                 |               |                   |
 |                 |               |  (19) 200 OK      |
 |                 |  (20) 200 OK  |<------------------|
 |   (21) 200 OK   |<--------------|                   |
 |<----------------|  (22) ACK     |                   |
 |---------------------------------------------------->|
 |                 |               |                   |
```

Signaling messages (1) to (3), placing the first call on hold, are identical to those used in Call Waiting (see Section 10.10), and are not reproduced here.

Signaling messages (4) to (22), placing the second call, are identical to those for a basic call flow (see Section 10.1), and are not reproduced here. For this example, assume the Call-ID was B64(SHA-1(555-1111;time=36124125;seq=23))@localhost.

```
State at MTA-o for call from MTA-o to MTA-t2
     From: sip:B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     To: tel:555-3333
     Call-ID: B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     Contact: sip: Host(mta-t2.provider)
     Remote-Party-ID: tel:+1-212-555-3333
     State: Host(dp-o.provider); state="{gate= Host(cmts-
             o.provider): 3612/3S10782, nexthop=sip:+1-212-555-
             3333;rn=+1-212-256-3333;npdi=yes@Host(dp-t2.provider),
             state="Host(dp- t2.provider);
             nexthop=sip:555-3333@Host(mta-t2.provider);
             gate=Host(cmts- t2.provider):4321/31S14621;
             orig-dest=tel:+1-212-555-1111; num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
             555-1111/212-555-3333>
     Dcs-Billing-ID: Host(dp-o.provider):3612E5C:0152
```

After some period of consultation, MTA-o initiates a transfer of the call from MTA-t1 to the new destination, MTA-t2. This involves placing the second call on hold (message sequence described earlier), and sending a REFER message to MTA-t2, giving it the information about the call with MTA-t1 in the REFER-By header. The INVITE message, since it changes parties involved in the call, is routed through the proxies. The sequence is shown in the following figure, and detailed below.


```
MTA-o    MTA-t1    Proxy-o      Proxy-t2              MTA-t2

 |  (1) REFER       |                |                    |
 |----------------->|  (2) REFER     |                    |
 |        |         |--------------->|   (3) REFER        |
 |        |         |                |------------------->|
 |        |                          |                    |
 |        |      Proxy-t1            | (4) INVITE         |
 |        |         |  (5)INVITE     |<-------------------|
 |        |(6)INVITE|<---------------|                    |
 |        |<--------|                |                    |
 |        | (7)SDP  |                |                    |
 |        |-------->|  (8) SDP       |                    |
 |        |         |--------------->|    (9) SDP         |
 |        |         |  (10) PRACK    |------------------->|
```

```
   |        |<--------------------------------------------|
   |        |         |  (11) 200 OK  |                   |
   |        |-------------------------------------------->|
   |        |         |  (12) COMET   |                   |
   |        |<--------------------------------------------|
   |        |         |  (13) 200 OK  |                   |
   |        |-------------------------------------------->|
   |        |(14)200  |               |                   |
   |        |-------->| (15) 200 OK   |                   |
   |        |         |-------------->|   (16) 200 OK     |
   |        |         |  (17) ACK     |------------------>|
   |        |<--------------------------------------------|
   |        |         |               |                   |
   |        |      Proxy-o            | (17) 200 OK       |
   |        |         |  (18)200 OK   |<------------------|
   |    (19) 200 OK   |<--------------|                   |
   |<-----------------|               |                   |
   |        |         |  (20) BYE     |                   |
   |--------------------------------------------------->|
   |        |         |  (21) 200 OK  |                   |
   |<-----------------------------------------------------|
   |(22)BYE |         |               |                   |
   |------->|         |               |                   |
   |(23)200 |         |               |                   |
   |<-------|         |               |                   |
```


After placing the second call on hold, MTA-o initiates a transfer by sending a REFER to MTA-t2, routed through the proxies.

(1) REFER:

```
     REFER sip: Host(mta-t2.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Supported: 100rel, state
     Remote-Party-ID: John Doe <tel:555-1111>
     Anonymity: off
     Refer-to: tel:+1-212-555-2222 ? Call-ID=B64(SHA-1(555-
             1111;time=36124033;seq=72) & Referred-by=tel:555-1111
             & State= Host(dp-o.provider);state="{nexthop=sip:Host(
             dp-t1.provider); gate=Host(cmts-
             o.provider):3612/17S30124; state="Host(dp-t1.provider);
             nexthop=sip:555-2222@Host(mta-t1.provider);
             gate=Host(cmts-t1.provider):4321/31S14621; orig-
             dest=tel:+1-212-555-1111; num-redirects=0"}K"
```

```
     State: Host(dp-o.provider); state="{gate= Host(cmts-
             o.provider): 3612/3S10782, nexthop=sip:+1-212-555-
             3333;rn=+1-212-256-3333;npdi=yes@Host(dp-t2.provider),
             state="Host(dp- t2.provider);
             nexthop=sip:555-3333@Host(mta-t2.provider);
             gate=Host(cmts- t2.provider):4321/31S14621;
             orig-dest=tel:+1-212-555-1111; num-redirects=0"}K"
     From: sip:B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     To: tel:555-3333
     Call-ID: B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     CSeq: 133 REFER
     Referred-by: sip:B64(SHA-1(555-1111;time=36124125;
             seq=23))@localhost
```

When the REFER is received at Proxy-o, it first verifies MTA-o has subscribed to Call Transfer service. If so, it decrypts the State information in the Refer-to header to determine the local gate location and identification. Proxy-o queries the gate to obtain the transferred call's original billing information. Proxy-o inserts billing information to indicate that the user associated with the number 212-555-1111 will pay for the new call segment. Proxy-o extracts the call routing from the Dcs-state information, and then forwards the message to Proxy-t2.

(2) REFER:

```
     REFER sip: Host(dp-o.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Supported: 100rel, state
     Proxy-Require: dcs
     State: Host(dp-t2.provider); nexthop=sip:555-3333@Host(mta-
             t2.provider); gate=Host(cmts-
             t2.provider):4321/31S14621; orig-dest=tel:+1-212-555-
             1111; num-redirects=0
     Refer-to: tel:+1-212-555-2222? Call-ID=B64(SHA-1(555-
             1111;time=36124033;seq=72) & Referred-by=tel:555-1111
```



```
             & Dcs-Billing-Info= Host(rks-t1.provider)<4278-9865-
             8765-9000/212-555-2222/212-555-1111> & Dcs-Billing-
             Info= Host(rks-t2.provider)<5123-0123-4567-8900/212-
             555-1111/212-555-3333> & Dcs-Billing-ID= Host(dp-
             o.provider): 36123E5C:0152
     Remote-Party-ID: John Smith <tel:+1-212-555-2222>
     From: sip:B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     To: tel:555-3333
     Call-ID: B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
```

```
     CSeq: 133 REFER
     Referred-by: sip:B64(SHA-1(555-1111;time=36124125;
             seq=23))@localhost
```

Proxy-t2 forwards the REFER message to MTA-t2 after encrypting the destination of the transfer, and the Dcs-Billing, Dcs-Billing-ID headers.

```
(3) REFER:
     REFER sip: 555-3333@Host(mta-t2.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t2.provider), {via="Host(dp-
             o.provider); branch=1"; via=Host(mta-o.provider)}K
     Supported: 100rel, state
     Refer-to:  sip:{type=transfer; dest=tel:+1-212-555-2222;
             billing-id=Host(dp-o.provider): 36123E5C:0152;
             expires=<timestamp>; billing-info= Host(rks-
             t1.provider)<4278-9865-8765-9000/212-555-2222/212-555-
             1111> ; billing-info= Host(rks-t2.provider)<5123-0123-
             4567-8900/212-555-1111/212-555-3333>}K@Host(dp-
             t2.provider);user=private ? Call-ID=B64(SHA-1(555-
             1111;time=36124033;seq=72) & Referred-by=tel:555-1111
     Remote-Party-ID: John Smith <tel:+1-212-555-2222>
     From: sip:B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     To: tel:555-3333
     Call-ID: B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     CSeq: 133 INVITE
     Referred-by: sip:B64(SHA-1(555-1111;time=36124125;
             seq=23))@localhost
```

After processing the REFER, MTA-t2 issues a INVITE to MTA-t1. In addition to the standard headers carried in an INVITE message, the encrypted {Dcs-Billing, Dcs-Billing-ID} fields received in the REFER message are copied into the Request-URI of the INVITE message. These fields indicate the destination, and that the user associated with the 212-555-1111 number will be charged for the second call leg.

```
(4) INVITE:
     INVITE sip:{type=transfer; dest=tel:+1-212-555-2222; billing-
             id=Host(dp-o.provider): 36123E5C:0152;
             expires=<timestamp>; billing-info= Host(rks-
             t1.provider)<4278-9865-8765-9000/212-555-2222/212-555-
             1111> ; billing-info= Host(rks-t2.provider)<5123-0123-
```



```
             4567-8900/212-555-1111/212-555-3333>}K@Host(dp-
             t2.provider);user=private SIP/2.0
```

```
     Via: SIP/2.0/UDP Host(mta-t2.provider)
     Supported: 100rel, state
     Remote-Party-ID: John Smith <tel:555-3333>
     Anonymity: Off
     From: "Alien Blaster" <sip:B64(SHA-1(555-3333; time=36124172;
             seq=74))@localhost>
     To: sip:B64(SHA-1(555-3333; time=36124172; seq=75))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36124033;seq=72)@localhost
     Cseq: 129 INVITE
     Referred-by: tel:555-1111
     Contact: sip:Host(mta-t2.provider)
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mta-t2.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     a=rtpmap:96 G726-32/8000
     m=audio 3456 RTP/AVP 0
     a=qos:mandatory sendrecv
     a=X-pc-codecs:96
```

When the Proxy-t2 receives the INVITE it first decrypts the header information to find the real destination for the call. Proxy compares the current time against the timestamp in the encrypted string; if the request is too old, it is refused. It invokes the call routing logic to determine which Proxy (Proxy-t1) to which the INVITE needs to be routed. It also embeds two Dcs-Billing-Info headers in this message. The first one identifies the user associated with the E.164 number 212-555-2222 as paying for the initial call leg (212-555-2222/212-555-1111). This information was derived from the customer account information for the caller during the first call attempt. The second Dcs-Billing-Info header identifies the user associated with the E.164 number 212-555-1111 as paying for the second call leg (212-555-1111/212-555-3333), and was provided by Proxy-o in the REFER message.

(5) INVITE:

```
     INVITE sip: +1-212-555-2222;rn=+1-212-265-2222;
             npdi=yes@Host(dp-t1);user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t2.provider); branch=1;
     Via: SIP/2.0/UDP Host(mta-t2.provider);
     Supported: 100rel, state
     Require: state
     Proxy-Require: dcs, state
```

```
Remote-Party-ID: John Smith <tel:+1-212-555-3333>
Anonymity: Off
Dcs-Gate: Host(cmts-t2.provider):3612/17S30124/37FA1948
Dcs-Billing-Info: Host(rks-t1.provider)<4278-9865-8765-
        9000/212-555-2222/212-555-1111>
Dcs-Billing-Info: Host(rks-t2.provider)<5123-0123-4567-
        8900/212-555-1111/212-555-3333>
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
From: "Alien Blaster" <sip:B64(SHA-1(555-3333; time=36124172;
        seq=74))@localhost>
To: sip:B64(SHA-1(555-3333; time=36124172; seq=75))@localhost
Call-ID: B64(SHA-1(555-1111;time=36124033;seq=72)@localhost
Cseq: 129 INVITE
Reffered-By: tel:555-1111
Contact: sip:Host(mta-t2.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE, Proxy-t1 queries the directory server to determine the IP address (MTA-t1) associated with 212-555-2222. It then forwards the INVITE message to MTA-t1, after stripping off all of the billing fields, and adding the encrypted state information. MTA-t1 recognizes the Call-ID matching an existing call, and matches the value of the Reffered-By: header to the From/To of that call. Since they match, the call is allowed to proceed, with the 183-Session-Progress, receiving PRACK, COMET, etc. These messages are identical to the basic call flow shown in Section 10.1, and are not repeated here.

Upon receipt of the 200-OK response from MTA-t1, MTA-t2 sends the final response of the REFER by sending a 200-OK to MTA-o. This message is routed through the Proxy Proxy-t2, Proxy-o, and then delivered to MTA-o. MTA-o responds directly with an ACK. Proxy-o is now done, and MTA-o sends a BYE message to terminate the call leg from MTA-o to MTA-t2, and a BYE message to terminate the call leg from MTA-o to MTA-t1.

10.13 Three-Way-Calling (with Network Bridge)

Three-way calling is a fairly complex consumer service that allows a subscriber to simultaneously talk to two parties, and for those two parties to hear each other. It is often thought of as an ad-hoc conference bridge. Usage of the service proceeds as follows. The customer has an active call, either one initiated or received. The customer then does a hookflash, which places the existing call on hold and presents a dialtone. The user then dials the a second number, and connects to that party. A hookflash at this point creates a 3-way call, bridging the two calls together. Note the distinction between three-way calling and call waiting (where the two calls are alternately placed on hold and connected) lies in the fact that the subscriber initiated the second call; if the second call was an incoming call then the call-waiting service would be active.

The desired state during the three-way-call is three separate call legs, from each participant to the bridge server. If the participants initiate the calls, then they all have the same Call-ID, which tells the bridge to mix them together. If the bridge initiates the connections, there is no necessity for a common Call-ID. Multiple methods exist using combinations of REFER methods and headers to achieve the desired connections. One way involves the subscriber establishing a connection to a bridge element, then transferring both of the existing calls to the bridge. Another method involves the subscriber asking the bridge to handle redirecting the existing calls to itself. The latter involves fewer signaling messages, and is preferred over the former. There is, of course, a third option - that the conference bridging function is done within the MTA and the network sees it as two separate simultaneous calls. As this consumes double the access network bandwidth, it is discouraged.

Initially a single call is active. For purposes of this example, consider that call to have been a call initiated by MTA-t1 to MTA-o. The call identification information at MTA-o is as follows:

```
MTA-o state for call from MTA-t1 to MTA-o
     From: sip:B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
     To: tel:555-1111
     Call-ID: B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
```



```
     Contact: sip: Host(mta-t1.provider)
     Remote-Party-ID: tel:+1-212-555-2222
     State: Host(dp-o.provider); state="{nexthop=sip:Host(dp-
             t.provider); gate=Host(cmts-o.provider):3612/17S30124;
             state="Host(dp-t.provider); nexthop=sip:555-
             2222@Host(mta-t1.provider); gate=Host(cmts-
             t.provider):4321/31S14621; orig-dest=tel:+1-212-555-
             1111; num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-t1.provider)<4278-9865-8765-
     9000/212-555-2222/212-555-1111>
     Dcs-Billing-ID: Host(dp-t1.provider):36124033:0381
```

MTA-o observes a hookflash and places this call on hold, issues a dialtone, and collects digits for a second call (212-555-3333). This sequence is shown in Section 10.10, resulting in the first call being held and a conversation active to the second destination. For this example, assume the second call is identified as follows:

```
State at MTA-o for call from MTA-o to MTA-t2
     From: sip:B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     To: tel:555-3333
     Call-ID: B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     Contact: sip: Host(mta-t2.provider)
     Remote-Party-ID: tel:+1-212-555-3333
     State: Host(dp-o.provider); state="{gate= Host(cmts-
             o.provider): 3612/3S10782, nexthop=sip:+1-212-555-
             3333;rn=+1-212-256-3333@Host(DP-t), state="Host(dp-
             t.provider); nexthop=sip:555-3333@Host(mta-t.provider);
             gate=Host(cmts-t.provider):4321/31S14621; orig-
             dest=tel:+1-212-555-1111; num-redirects=0"}K"
```

```
      Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
      555-1111/212-555-3333>
      Dcs-Billing-ID: Host(dp-o.provider):3612E5C:0152
```

The three-way-calling method described in this appendix asks the bridge to redirect the existing calls via an INVITE(Refer). The bridge therefore is in control of managing the endpoints, and knows the proper media streams for mixing, even though they don't have a common Call-ID.


```
   MTA-o    MTA-t1    Proxy-o      Proxy-b               MTA-t2

     |                  |  (1) INVITE(Hold)                |
     |------------------------------------------------------>|
     |                  |  (2) 200 OK   |                    |
     |------------------------------------------------------>|
     |                  |  (3) ACK      |                    |
     |------------------------------------------------------>|
     |                  |               |
     |  (4) REFER      |               |                Bridge
     |----------------->|  (5) REFER   |                   |
     |        |         |-------------->|   (6) REFER      |
     |        |         |               |------------------->|
```

```
    |        |         |               |  (7) 183 SDP      |
    |        |         |  (8) 183 SDP  |<------------------|
    |   (9) 183 SDP    |<--------------|                   |
    |<-----------------| (10) PRACK    |                   |
    |----------------------------------------------------->|
    |        |         | (11) 200 OK   |                   |
    |<-----------------------------------------------------|
    |        |         | (12) COMET    |                   |
    |----------------------------------------------------->|
    |        |         | (13) 200 OK   |                   |
    |<-----------------------------------------------------|
    |        |         |               | (14) 200 OK       |
    |        |         | (15) 200 OK   |<------------------|
    |   (16) 200 OK    |<--------------|                   |
    |<-----------------| (17) ACK      |                   |
    |----------------------------------------------------->|
    |        |         |               |                   |
    |        |         |               | (18) INVITE       |
    |        |         |  (19)INVITE   |<------------------|
    |        (20)INVITE|<--------------|                   |
    |        |<--------|               |                   |
    |        |(21)SDP  |               |                   |
    |        |-------->| (22) SDP      |                   |
    |        |         |-------------->|   (23) SDP        |
    |        |         |  (24) PRACK   |------------------>|
    |        |<--------------------------------------------|
    |        |         |  (25) 200 OK  |                   |
    |        |-------------------------------------------->|
    |        |         |  (26) COMET   |                   |
```



```
    |        |<--------------------------------------------|
    |        |         |  (27) 200 OK  |                   |
    |        |-------------------------------------------->|
    |        |(28)200  |               |                   |
    |        |-------->| (29) 200 OK   |                   |
    |        |         |-------------->|   (30) 200 OK     |
    |        |         |  (31) ACK     |------------------>|
    |(32)BYE |<--------------------------------------------|
    |<-------|         |               |                   |
    |(33)200 |         |               |                   |
    |------->|         |               |                   |
```

Messages (1) to (3), for putting an existing call on hold, are identical to those used in Call Waiting (see Section 10.10).
In response to the hook-flash, MTA also issues an INVITE to a bridge with a new call ID. The identity of the destination is given via

the service name "bridge," which is a pre-defined service name in DCS.

(4) REFER:

```
REFER sip: bridge@Host(dp-o) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
Supported: 100rel, state
Remote-Party-ID: John Doe <tel:555-1111>
Anonymity: Off
Refer-To: tel:+1-212-555-2222 ? Call-ID=B64(SHA-1(555-
        2222;time=36124033;seq=72))@localhost & Reffered-By
        =tel:555-1111 & State= Host(dp-o.provider);
        state="{nexthop=sip:Host(dp-t.provider);
        gate=Host(cmts-o.provider):3612/17S30124;
        state="Host(dp-t.provider); nexthop=sip:555-
        2222@Host(mta-t1.provider); gate=Host(cmts-
        t.provider):4321/31S14621; orig-dest=tel:+1-212-555-
        1111; num-redirects=0"}K"
Refer-To: tel:+1-222-555-3333 ? Call-ID=B64(SHA-1(555-
        1111;time=36124125;seq=23))@localhost & Reffered-By
        =B64(SHA-1(555-1111;time-
        36124125;seq=23))@localhost & State= Host(dp-
        o.provider); state="{gate= Host(cmts-o.provider):
        3612/3S10782, nexthop=sip:+1-212-555-3333;rn=+1-212-
        256-3333@Host(DP-t), state="Host(dp-t.provider);
        nexthop=sip:555-3333@Host(mta-t.provider);
        gate=Host(cmts-t.provider):4321/31S14621; orig-
        dest=tel:+1-212-555-1111; num-redirects=0"}K"
From: sip:B64(SHA-1(555-1111; time=36124135;seq=24))@localhost
To: sip: bridge@Host(dp-o.provider)
Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
Contact: sip:Host(mta-o.provider)
Cseq: 131 INVITE
Content-Type: application/sdp
Content-length: (.)

v=0
```



```
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
```

```
m=audio 3460 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Proxy-o resolves the "bridge service name" to an available bridge (mcu41@Host(dp-b.provider) in this example), and forwards the INVITE to the associated Proxy (Proxy-b). In general, bridges will be available locally at Proxy-o, but this example demonstrates the messages exchanged if the bridge is remote. In general, bridges will be network services and located within the trusted domain of the network. However, they may also be provided by others. This example call flow diagram shows the latter case, where the bridge is outside the trusted domain of the service provider.

If the bridge is a trusted network element, the Bridge (for signaling purposes) would be functionally equivalent to a CMS, and use the same message set as is used between CMSs. In the diagram above, this would appear as if the lines Proxy-b and BRIDGE were merged together.

Proxy-o decrypts the state header values attached to the Refer-To headers, extracts the billing information for each of the previous call legs, and expands this information into the Dcs-Billing-Info values

(5) REFER:

```
REFER sip:mcu41@Host(dp-b.provider) SIP/2.0
Via: SIP/2.0/UDP Host(dp-o.provider)
Via: SIP/2.0/UDP Host(mta-o.provider)
Supported: 100rel, state
Proxy-Require: dcs, state
Require: state
Remote-Party-ID: John Doe <tel:+1-212-555-1111>
Anonymity: Off
Dcs-Gate: Host(cmts-o.provider):3612/5S12045/9142E7A1
Dcs-Billing-Info: Host(rks-o.provider)<5123-4567-8900/212-555-
        1111/mcu41@Host(dp-b.provider)/bridge-3>
Dcs-Billing-ID: Host(dp-o.provider):36124135:92
Refer-To: tel:+1-212-555-2222 ? CallID= B64(SHA-1(555-
        2222;time=36124033;seq=72))@localhost & Dcs-Billing-
        Info= Host(rks-t1.provider)<4278-9865-8765-9000/212-
        555-2222/212-555-1111> & Dcs-Billing-Info= Host(rks-
        o.provider)<5123-4567-8900/212-555-1111/mcu41@Host(dp-
        b.provider)> & Dcs-Billing-ID= Host(dp-
        t1.provider):36124033:0381 & Reffered-By=tel:555-1111
```

```
     Refer-to: tel:+1-212-555-3333 ? CallID= B64(SHA-1(555-
             1111;time=36124125;seq=23))@localhost & Dcs-Billing-
             Info= Host(rks-o.provider)/<5123-4567-8900/212-555-
             1111/212-555-3333> & Dcs-Billing-Info= Host(rks-
             o.provider)<5123-4567-8900/212-555-1111/mcu41@Host(dp-
             b.provider)> & Dcs-Billing-ID= Host(dp-
             o.provider):36123E5C:0152 & Reffered-By:B64(SHA-1(555-
             1111;time-36124125;seq=23))@localhost
     State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
             o.provider); gate=Host(cmts-o.provider):3612/17S30124
     From: sip:B64(SHA-1(555-1111; time=36124135;seq=24))@localhost
     To: sip: bridge@Host(dp-o.provider)
     Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
     Contact: sip:Host(mta-o.provider)
     Cseq: 131 INVITE
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mta-o.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     A=rtpmap:0 PCMU/8000
     a=rtpmap:96 G726-32/8000
     m=audio 3460 RTP/AVP 0
     a=qos:mandatory sendrecv
     a=X-pc-codecs:96
```

Proxy-b encrypts the various fields into two Refer-To: headers, caches the Via headers, and passes the message to the bridge.

(6) REFER:

```
     REFER sip:mcu41.provider SIP/2.0
     Via: SIP/2.0/UDP Host(dp-b.provider), {via=Host(dp-o.provider);
             via=Host(mta-o.provider)}K
     Supported: 100rel, state
     Require: state
     Remote-Party-ID: John Doe <tel:+1-212-555-1111>
     Dcs-Gate: 27S6028
     Refer-To: sip:{type=transfer; dest=+1-212-555-2222; Billing-
             info=Host(dp-t1.provider):36124033:0381; <timestamp>;
             Billing-info=Host(rks-t1.provider)<4278-9865-8765-
             9000/212-555-2222/212-555-1111>; Billing-id=Host(rks-
             o.provider)<5123-4567-8900/212-555-
             1111/mcu41.provider>}K@dp-b.provider;user=private ?
             Call-ID= B64(SHA-1(555-2222;time=36124033;seq=72))
             @localhost & Reffered-By=tel:555-1111
     Refer-To: sip:{type=transfer; dest=+1-212-555-3333; billing-
```

```
            id=Host(dp-o.provider):36123E5C:0152;
            expires=<timestamp>; billing-info=Host(rks-
```



```
            o.provider)/<5123-4567-8900/212-555-1111/212-555-3333>;
            billing-info=Host(rks-o.provider)<5123-4567-8900/212-
            555-1111/mcu41.provider>}K@dp-b.provider; user=private
    ?       Call-ID= B64(SHA-1(555-
            1111;time=36124125;seq=23))@localhost & Reffered-By
            :B64(SHA-1(555-1111;time-36124125;seq=23))
    State: Host(dp-b.provider); state="{nexthop=sip:Host(dp-
            o.provider); gate=Host(cmts-b.provider): 3612/27S6028;
            via="Host(dp-o.provider);branch=1", via=Host(mta-
            o.provider), state="Host(dp-o.provider);
            nexthop=sip:555-1111@Host(mta-o.provider);
            gate=Host(cmts-o.provider):3612/17S30124"}K"
    From: sip:B64(SHA-1(555-1111; time=36124135;seq=24))@localhost
    To: sip: bridge@Host(dp-o.provider)
    Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
    Contact: sip:Host(mta-o.provider)
    Cseq: 131 INVITE
    Content-Type: application/sdp
    Content-length: (.)

    v=0
    o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
    s=-
    c= IN IP4 Host(mta-o.provider)
    b=AS:64
    t=907165275 0
    a=X-pc-csuites:312F
    a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
    A=rtpmap:0 PCMU/8000
    a=rtpmap:96 G726-32/8000
    m=audio 3460 RTP/AVP 0
    a=X-pc-codecs:96
```

Bridge completes the call from MTA-o in a manner very similar to the basic call flow of Section 10.1. Since a bridge doesn't need to alert a human, it responds immediately with 200-OK when resources are known to be available. Messages (7) through (17) are not detailed in this section.

Bridge initiates two calls in parallel, one to each of the participants listed in the Refer-To: headers. The Request URI in the new INVITE message is the encrypted string received in the Refer-To: header, the To: header is a generic string such as "participant<n>" since the bridge has no knowledge of the identity of the participants, and the Call-ID is the value from the Refer-To header. Part of the message sequence for MTA-t1 (messages (18) to (33)) is detailed here; messages (34) through (49) are identical and not shown in the figure.

```
(18) INVITE (Refer-By):
     INVITE sip:{type=transfer; dest=+1-212-555-2222; Billing-
             info=Host(dp-t1.provider):36124033:0381; <timestamp>;
             Billing-info=Host(rks-t1.provider)<4278-9865-8765-
```



```
             9000/212-555-2222/212-555-1111>; Billing-id=Host(rks-
             o.provider)<5123-4567-8900/212-555-
             1111/mcu41.provider>}K@dp-b.provider; user=private
             SIP/2.0
     Via: SIP/2.0/UDP Host(mcu41.provider)
     Supported: 100rel, state
     Remote-Party-ID: Bridge Service <sip:Host(mcu41.provider)>
     Anonymity: URL, Name
     From: sip:B64(SHA-1(bridge;time=36124135;seq=311))@localhost
     To: sip: participant1@localhost
     Call-ID: B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
     Contact: sip:Host(mcu41.provider)
     Cseq: 128 INVITE
     Refer-By:tel:555-1111
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mcu41.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     a=rtpmap:96 G726-32/8000
     m=audio 3174 RTP/AVP 0
     a=X-pc-Qos:mandatory sendrecv
     a=X-pc-codecs:96
```

Proxy-b decodes the encrypted Request-URI to find the real destination for this call. Proxy compares the current time against the expiration time in the encrypted string; if the request is too old it is refused. The call routing is determined from the destination contained in the encrypted string, as is the billing information for the call. Proxy-b sends the following INVITE message to Proxy-t1:

```
(19) INVITE (Refer-By):
     INVITE sip:+1-212-555-2222;rn=+1-212-234-2222;
             npdi=yes@Host(dp-t1.provider);user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-b.provider)
     Via: SIP/2.0/UDP Host(mcu41.provider)
     Supported: 100rel, state
     Require: state
     Proxy-Require: dcs, state
     Remote-Party-ID: Bridge Service <sip:Host(mcu41.provider)>
     Anonymity: URL, Name
     Dcs-Gate: Host(cmts-b.provider):3612/28S6029/079317A3
     State: Host(dp-b.provider); nexthop=Host(mcu41.provider);
             gate=Host(cmts-b)::3621/28S6029; orig-dest=tel:+1-212-
             555-2222; num-redirects=0
```



```
     From: sip:B64(SHA-1(bridge;time=36124135;seq=311))@localhost
     To: sip: participant1@localhost
     Call-ID: B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
     Contact: sip:Host(mcu41.provider)
     Cseq: 128 INVITE
     Refer-By:tel:555-1111
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(bridge.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     a=rtpmap:96 G726-32/8000
     a=qos:mandatory sendrecv
     m=audio 3174 RTP/AVP 0
```

Proxy-t1 processes this exactly as a normal INVITE message, and passes the message to MTA-t1.

```
(20) INVITE (Refer-By):
     INVITE sip:555-2222@Host(mta-t1.provider) SIP/2.0
```

```
Via: SIP/2.0/UDP Host(dp-t1.provider), {via=Host(dp-
        b.provider); via=Host(mcu41.provider)}K
Supported: 100rel, state
Require: state
Remote-Party-ID: <sip:{type=rem-id;dest=sip:Host(mcu41.
        provider)}K@Host(dp-t1.provider); user=private>
Media-Authorization: 5S32740
State: Host(dp-t1.provider); state={nexthop=sip:Host(dp-b);
        gate=Host(cmts-t1:3621/53S32740;state="Host(dp-
        b.provider); nexthop=Host(mcu41.provider);
        gate=Host(cmts-b)::3621/28S6029; orig-dest=tel:+1-212-
        555-2222; num-redirects=0"}K
From: sip:B64(SHA-1(bridge;time=36124135;seq=311))@localhost
To: sip: participant1@localhost
Call-ID: B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
Contact: sip:Host(mcu41.provider)
Cseq: 128 INVITE
Refer-By:tel:555-1111
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mcu41.provider)
b=AS:64
```



```
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3174 RTP/AVP 0
a=X-pc-Qos:mandatory sendrecv
a=X-pc-codecs:96
```

MTA-t1 notes the Call-ID: header, and determines that it has a call with that ID and that the Refer-By: header matches either the From: or To: value for the call. This INVITE is therefore interpreted as an update to that existing call. The provisional response (183-Session-Progress) is sent (21) to the local Proxy, who restores the encrypted Via: headers and sends it (22) to the originating Proxy, who passes it (23) to the bridge. These messages are identical to those of the basic call flow. The bridge responds with the PRACK message (24), as in the basic call flow. The bridge then performs the resource allocation and continues as in the basic call flow.

Upon receipt of the ACK message, MTA-t1 sends a BYE message to its original caller. Note that this message has the From: and To: headers reversed from the incoming INVITE originally received for this call. If MTA-t1 had initiated the call to MTA-o, then the From: and To: would match those in the initial INVITE.

(32) BYE:

```
BYE sip:Host(mta-o.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-t1.provider)
From: tel:555-1111
To: sip:B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
Call-ID: B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
Cseq: 129 BYE
```

Upon receipt of the BYE message, MTA-o sends the following 200-OK message to MTA-t1.

(33) 200-OK:

```
SIP/2.0  200 OK
Via: SIP/2.0/UDP Host(mta-t1.provider)
From: tel:555-1111
To: sip:B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
Call-ID: B64(SHA-1(555-2222;time=36124033;seq=72))@localhost
Cseq: 129 BYE
```

The sequence of messages (34)-(49) is identical, and performs the same functions for the other leg of the three-way conference.

10.14 Three-Way-Calling Hangup sequences

There are two distinct hangup sequences that need to be detailed: hangup of a participant and hangup of the originator. The first results in a basic call between the originator and the remaining participant. The latter results in a hangup of all participants.

For both of the following detail call flows, consider the initial state information to be the following:

```
MTA-o: Call from MTA-o to BRIDGE
     From: sip:B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
     To: sip:bridge@Host(dp-o.provider)
     Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
     Contact: sip:Host(mcu41.provider)
     State: Host(dp-o.provider); state="{gate= Host(cmts-
             o.provider): 3612/12S52127, nexthop=sip: mcu41@Host(dp-
             b), state="Host(dp-b.provider);
             nexthop=sip:Host(mcu41.provider); gate=Host(cmts-
             b.provider):4321/31S14621; orig-dest=sip:mcu41@Host(dp-
             b); num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-o.provider)/341FE8B<5123-0123-4567-
             8900/212-555-1111/mcu41.provider/Bridge-3

MTA-t1: Call from BRIDGE to MTA-t1
     From: sip:B64(SHA-1(bridge;time=36124135;seq=311))@localhost
     To: sip: participant1@localhost
     Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24)) @localhost
     Contact: sip:Host(mcu41.provider)
     State: Host(dp-t1.provider); state="{nexthop=sip:Host(dp-
             b.provider); gate=Host(cmts-t1.provider):
             3612/12S52127; state="Host(dp-b.provider);
             nexthop=sip:Host(mcu41.provider); gate=Host(cmts-
             b.provider):3612/17S30124; orig-dest=tel:+1-212-555-
             2222; num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-t1.provider)<4278-9865-8765-
             9000/212-555-2222/212-555-1111>;Host(rks-
             o.provider)<5123-4567-8900/212-555-2222/mcu41.provider>


MTA-t2: Call from BRIDGE to MTA-t2
     From: sip:B64(SHA-1(bridge;time=36124135;seq=312)) @localhost
```



```
     To: sip: participant2@localhost
     Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24)) @localhost
     Contact: sip:Host(mcu41.provider)
     State: Host(dp-t2.provider); state="{nexthop=sip:Host(dp-
             b.provider); gate=Host(cmts-t2.provider):
```

```
                3612/13S52196; state="Host(dp-b.provider);
                nexthop=sip:Host(mcu41.provider); gate=Host(cmts-
                b.provider):3612/18S37224; orig-dest=tel:+1=212-555-
                3333; num-redirects=0"}K"
        Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
                555-1111/212-555-3333>;Host(rks-o.provider)<5123-4567-
                8900/212-555-3333/mcu41.provider>

   Bridge: Call from MTA-o to BRIDGE
        From: sip:B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
        To: sip:bridge@Host(dp-o.provider)
        Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
        Contact: sip:Host(mta-o.provider)
        Remote-Party-ID: tel:+1-212-555-1111
        State: Host(dp-b.provider); state="{nexthop=sip:Host(dp-
                o.provider); gate=Host(cmts-b.provider):3612/15S30179;
                state="Host(dp-o.provider); nexthop=sip:555-
                1111@Host(mta-o.provider); gate=Host(cmts-
                o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
                1111; num-redirects=0"}K"
        Dcs-Billing-Info: Host(rks-o.provider)/341FE8B<5123-0123-4567-
                8900/212-555-1111/mcu41.provider/Bridge-3

   Bridge: Call from BRIDGE to MTA-t1
        From: sip:B64(SHA-1(bridge;time=36124135;seq=311))@localhost
        To: sip: participant1@localhost
        Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
        Contact: sip:Host(mta-t1.provider)
        State: Host(dp-b.provider); state="{gate= Host(cmts-
                b.provider): 3612/17S30124, nexthop=sip:+1-212-555-
                2222;rn=+1-212-234-2222@Host(DP-t1), state="Host(dp-
                t1.provider); nexthop=sip:555-2222@Host(mta-
                t1.provider); gate=Host(cmts-
                t1.provider):4321/31S14621; orig-dest=tel:+1-212-555-
                2222; num-redirects=0"}K"
        Dcs-Billing-Info: Host(rks-t1.provider)<4278-9865-8765-
                9000/212-555-2222/212-555-1111>;Host(rks-
                o.provider)<5123-4567-8900/212-555-2222/mcu41.provider>

   Bridge: Call from BRIDGE to MTA-t2
        From: sip:B64(SHA-1(bridge;time=36124135;seq=312))@localhost
        To: sip: participant2@localhost
        Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
        Contact: sip:Host(mta-t2.provider)
        State: Host(dp-b.provider); state="{gate= Host(cmts-
                b.provider): 3612/18S37624, nexthop=sip:+1-212-555-
                3333;rn=+1-212-234-3333@Host(DP-t2), state="Host(dp-
                t2.provider); nexthop=sip:555-3333@Host(mta-
```

```
             t2.provider); gate=Host(cmts-
             t2.provider):3621/13S52196; orig-dest=tel:+1-212-555-
             3333; num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
             555-1111/212-555-3333>;Host(rks-o.provider)<5123-4567-
             8900/212-555-3333/mcu41.provider>
```

```
MTA-o    MTA-t1    Proxy-o      Proxy-b               Bridge

 |        |         |  (1) BYE      |                   |
 |        |-------------------------------------------->|
 |        |         |  (2) 200 OK   |                   |
 |        |<--------------------------------------------|
 |        |         |               |                   |
 |        |         |               |  (3) REFER       |
 |        |         |  (4) REFER   |<------------------|
 |   (5) REFER     |<--------------|                   |
 |<-----------------|               |                   |
 |  (6) 200 OK      |               |                   |
 |----------------->|  (7) 200 OK   |                   |
 |        |         |-------------->|   (8) 200 OK      |
 |        |         |  (9) ACK      |------------------>|
 |<-----------------------------------------------------|
 |        |         |
 |        |         |            Proxy-t2            MTA-t2
 |        |         |               |                   |
 |  (10) INVITE     |               |                   |
 |----------------->|  (11) INVITE  |                   |
 |        |         |-------------->|  (12) INVITE      |
 |        |         |               |------------------>|
 |        |         |               | (13) 183 SDP      |
 |        |         | (14) 183 SDP  |<------------------|
 |   (15) 183 SDP   |<--------------|                   |
 |<-----------------| (16) PRACK    |                   |
 |----------------------------------------------------->|
 |        |         |               |                   |
```

For this example, consider MTA-t1 to drop out of the three-way conference. MTA-t1 sends a BYE message to terminate its current call.

(1) BYE

```
     SIP/2.0 BYE Host(mcu41.provider)
     Via: SIP/2..0/UDP Host(mta-t1.provider)
     From: sip:B64(SHA-1(bridge;time=36124135;seq=311))@localhost
     To: sip: participant1@localhost
```

```
Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24)) @localhost
CSeq: 12002 BYE
```

The Bridge responds to MTA-t1 with the expected acknowledgement message.

(2) 200-OK

```
SIP/2.0 200 OK
Via: SIP/2..0/UDP Host(mta-t1.provider)
From: sip:B64(SHA-1(bridge;time=36124135;seq=311))@localhost
To: sip: participant1@localhost
Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24)) @localhost
CSeq: 12002 BYE
```

The Bridge now reconnects the two remaining parties into a simple call. Since MTA-o is the originator of the three-way-call, the Bridge informs it of the need to redirect the call from MTA-t2. Bridge sends the INVITE(Also,Replace) message, via Proxy-b.

(3) REFER

```
REFER sip:Host(mta-o.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mcu41.provider)
Supported: 100rel, state
Refer-To: tel:+1-212-555-3333 ? Call-ID= B64(SHA-1(555-
        1111;time=36124135;seq=24))@localhost & Reffered-By=
        sip:B64(SHA-1(bridge;time=36124135;seq=312))@localhost
        & State= Host(dp-b.provider); state="{gate= Host(cmts-
        b.provider): 3612/18S37624, nexthop=sip:+1-212-555-
        3333;rn=+1-212-234-3333@Host(DP-t2), state="Host(dp-
        t2.provider); nexthop=sip:555-3333@Host(mta-
        t2.provider); gate=Host(cmts-
        t2.provider):3621/13S52196; orig-dest=tel:+1-212-555-
        3333; num-redirects=0"}K"
State: Host(dp-b.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-b.provider):3612/15S30179;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        1111; num-redirects=0"}K"
From: sip:bridge@Host(dp-o.provider)
To: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
CSeq: 12301 INVITE
Refer-By: sip:bridge@Host(dp-o.provider)
```

The call flow from this point onward is identical to the Call
Transfer with Consultation, as shown in Section 10.12. The bridge,
having "consulted" with MTA-o, transfers its call with MTA-t2 to
MTA-o.


```
MTA-o    MTA-t1    MTA-t2        Proxy-b              Bridge

 |        |         |  (1) BYE      |                   |
 |------------------------------------------------------>|
 |        |         |  (2) 200 OK   |                   |
 |<------------------------------------------------------|
 |        |         |               |                   |
 |        |         |  (3) BYE      |                   |
 |        |         |<-----------------------------------|
 |        |         |  (4) 200 OK   |                   |
 |        |         |----------------------------------->|
 |        |         |               |                   |
 |        |         |  (5) BYE      |                   |
 |        |<--------------------------------------------|
 |        |         |  (6) 200 OK   |                   |
 |        |--------------------------------------------->|
 |        |         |               |                   |
```

When the originator of a three-way call hangs up, the entire call is
terminated. The bridge recognizes the BYE from the originator and
sends BYE messages to all participants.

MTA-o -> Bridge

(1) BYE

```
      BYE sip:Host(mcu41.provider) SIP/2.0
      Via: SIP/2.0/UDP Host(mta-o.provider)
      From: sip:B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
      To: sip:bridge@Host(dp-o.provider)
      Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
      CSeq: 141  BYE

   Bridge -> MTA-o

   (2) 200-OK
      SIP/2.0 200 OK
      Via: SIP/2.0/UDP Host(mta-o.provider)
      From: sip:B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
      To: sip:bridge@Host(dp-o.provider)
      Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
      Cseq: 141  BYE

   Bridge -> MTA-t2
```



```
   (3) BYE
      BYE sip:Host(mta-t2.provider) SIP/2.0
      Via: SIP/2.0/UDP Host(mcu41.provider)
      From: sip:B64(SHA-1(bridge;time=36124135;seq=312))@localhost
      To: sip: participant2@localhost
      Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
      CSeq: 80001  BYE

   MTA-t2 -> Bridge

   (4) 200-OK
      SIP/2.0 200 OK
      Via: SIP/2.0/UDP Host(mcu41.provider)
      From: sip:B64(SHA-1(bridge;time=36124135;seq=312))@localhost
      To: sip: participant2@localhost
      Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
      CSeq: 80001  BYE

   Bridge -> MTA-t1

   (5) BYE
      BYE sip:Host(mta-t1.provider) SIP/2.0
      Via: SIP/2.0/UDP Host(mcu41.provider)
      From: sip:B64(SHA-1(bridge;time=36124135;seq=311))@localhost
      To: sip: participant1@localhost
      Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
      CSeq: 80002  BYE
```

```
MTA-t1 -> Bridge

(6) 200-OK
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(mcu41.provider)
     From: sip:B64(SHA-1(bridge;time=36124135;seq=311))@localhost
     To: sip: participant1@localhost
     Call-ID: B64(SHA-1(555-1111;time=36124135;seq=24))@localhost
     CSeq: 80002  BYE
```


10.15 CODEC Change within previous authorization

When the initial INVITE SDP contained multiple CODECs, such that any single CODEC would be authorized, no further interaction is needed with the CMS/Proxies to change CODECs. However, due to the requirements of the segmented reservation model of D-QoS, it is necessary to signal to the far end and synchronize changes in CODEC usage. The following diagram shows the sequence of signaling messages to perform this function.

```
MTA-o             Proxy-o        Proxy-b              MTA-t

 |                  |  (1) INVITE   |                   |
```

```
     |---------------------------------------------------->|
     |                  |  (2) 183 SDP  |                  |
     |<----------------------------------------------------|
     |                  |  (3) PRACK    |                  |
     |---------------------------------------------------->|
     |                  |  (4) 200 OK   |                  |
     |<----------------------------------------------------|
     |                  |  (5) COMET    |                  |
     |---------------------------------------------------->|
     |                  |  (6) 200 OK   |                  |
     |<----------------------------------------------------|
     |                  |               |                  |
     |                  |  (7) 200 OK   |                  |
     |<----------------------------------------------------|
     |                  |  (8) ACK      |                  |
     |---------------------------------------------------->|
     |                  |               |                  |
```

By some mechanism outside the scope of the Distributed Call Signaling protocol, MTA-o decides that a CODEC change is necessary. MTA-o sends the following INVITE message directly to MTA-t. This INVITE is almost identical to the initial INVITE that established the call, except for header fields such as Remote-Party-ID and Anonymity that are not sent to maintain originator privacy.

(1) INVITE:

```
     INVITE sip:Host(mta-t.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Supported: 100rel, state
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost >
```



```
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     CSeq: 129 INVITE
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mta-o.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
```

```
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
```

Upon receipt of the (1) INVITE message, MTA-t verifies its ability to switch to the designated CODEC, and responds with a 183-Session-Progress including an updated SDP. The procedures from this point onward are identical to a basic call flow, as given in Section 10.1.

The resource reservations done in this call flow maintain the previous resources, so that if either end fails to make the proper reservation, the original call can proceed with the initial CODEC. MTA-t actually switches to the new codec upon sending the final 200-OK response to the INVITE, and MTA-o switches to the new codec upon receipt of the final 200-OK.

10.16 CODEC Change requiring new authorization

When an MTA wishes to change to a different CODEC, but that CODEC was not among those initially authorized (or subsequently authorized by this sequence), it is necessary to request an increased authorization from the Proxy. The following diagram shows the sequence of signaling messages that achieves this.

```
MTA-o              Proxy-o       Proxy-t2   MTA-t1     MTA-t2

 |  (1) INVITE      |               |        |          |
 |----------------->|  (2) INVITE   |        |          |
 |                  |-------------->|   (3) INVITE      |
 |                  |               |------------------>|
 |                  |               |  (4) 183 SDP      |
 |                  |  (5) 183 SDP  |<------------------|
 |   (6) 183 SDP    |<--------------|        |          |
 |<-----------------|  (7) PRACK    |        |          |
 |----------------------------------------------------->|
 |                  |  (8) 200 OK   |                   |
 |<-----------------------------------------------------|
 |                  |  (9) COMET    |                   |
 |----------------------------------------------------->|
 |                  |  (10) 200 OK  |                   |
 |<-----------------------------------------------------|
 |                  |               |  (11) 200 OK      |
 |                  |  (12) 200 OK  |<------------------|
 |   (13) 200 OK    |<--------------|                   |
 |<-----------------|  (14) ACK     |                   |
 |----------------------------------------------------->|
 |                  |               |                   |
```

By some mechanism outside the scope of the Distributed Call Signaling protocol, MTA-o decides that a CODEC change is necessary, and that the previous authorization for the current call does not permit this new CODEC (e.g. the initial call setup used only G.726-32 and the new CODEC desired is G.711). MTA-o generates the following SIP INVITE message and sends it to Proxy-o (the Proxy that manages MTA-o). MTA-o starts timer (T-proxy-request).

(1) INVITE:

```
     INVITE sip:555-2222@Host(DP-o);user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
```



```
     Supported: 100rel, state
     State: Host(dp-o.provider); state="{gate= Host(cmts-
```

```
          o.provider): 3612/17S30124, nexthop=sip:+1-212-555-
          2222;rn=+1-212-234-2222@Host(DP-t), state="Host(dp-
          t.provider); nexthop=sip:555-2222@Host(mta-t.provider);
          gate=Host(cmts-t.provider):4321/31S14621; orig-
          dest=tel:+1-212-555-1111; num-redirects=0"}K"
  Remote-Party-ID: John Doe <tel:555-1111>
  From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
          seq=72))@localhost>
  To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
  Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
  CSeq: 129 INVITE
  Content-Type: application/sdp
  Content-length: (.)

  v=0
  O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
  s=-
  c= IN IP4 Host(mta-o.provider)
  b=AS:64
  t=907165275 0
  a=X-pc-csuites:312F
  a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
  a=X-pc-csuites:312F
  a=rtpmap:0 PCMU/8000
  m=audio 3456 RTP/AVP 0
  a=qos: mandatory sendrecv
```

Upon receiving the INVITE message, Proxy-o authenticates MTA-o using standard IPSec authentication. Proxy-o decodes the state string in the State header and extracts the relevant information about the current call. Proxy-o generates the following INVITE message and sends it to Proxy-t. Proxy-o adds a number of parameters to the INVITE message. These are shown below.

(2) INVITE:

```
  INVITE sip: +1-212-555-2222;rn=+1-212-234-2222;
          npdi=yes@Host(DP-t);user=phone SIP/2.0
  Via: SIP/2.0/UDP Host(DP-o.provider);branch=1
  Via: SIP/2.0/UDP Host(mta-o.provider)
  Supported: 100rel, state
  Require: state
  Proxy-Require: dcs, state
  Dcs-Gate: Host(cmts-o.provider): 3612/17S30124
  Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
          555-1111/212-555-2222>
  Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
  State: Host(dp-t.provider); nexthop=sip:555-2222@Host(mta-
          t.provider); gate=Host(cmts-t.provider):4321/31S14621;
          orig-dest=tel:+1-212-555-1111; num-redirects=0
  Remote-Party-ID: John Doe <tel:+1-212-555-1111>
```

```
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 INVITE
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mta-o.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     m=audio 3456 RTP/AVP 0
     a=qos: mandatory sendrecv
```

Upon receiving this INVITE message, Proxy-t recognizes this as a mid-call change by the presence of the State header with its name attached, and generates the following INVITE message and sends it to MTA-t. Note that the Via lines may be different from the initial INVITE exchange; they have been encrypted to maintain the privacy of the caller.

(3) INVITE:

```
     INVITE sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider);branch=1"; via=Host(mta-o.provider)}K
     Supported: 100rel, stateMedia-Authorization: 31S14621
     Remote-Party-ID: John Doe <tel:+1-212-555-1111>
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 INVITE
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mta-o.provider)
```

```
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
a=qos: mandatory sendrecv
```


Upon receiving this INVITE, MTA-t authenticates that the message came from Proxy-t using IPSec. MTA-t checks for a current call matching the triple (From, To, Call-ID). MTA-t looks at the capability parameters in the Session Description Protocol (SDP) part of the message and determines which media channel parameters it can accommodate for this call. MTA-t generates the following 183-Session-Progress response, and sends it to Proxy-t.

(4) 183 Session-Progress:

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider);branch=1"; via=Host(mta-o.provider)}K
Require: 100rel
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
Remote-Party-ID: John Smith <tel: 555-2222>
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 INVITE
Content-Disposition: precondition
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
```

```
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos: mandatory sendrecv confirm
```

Upon receiving the 200-OK message, Proxy-t authorizes the resources and forwards the following 183-Session-Progress to Proxy-o, restoring the Via headers. At this point Proxy-t has completed its transaction and does not maintain any more state for this call. Proxy-t may include Dcs-Billing-Information if it wishes to override the billing information that came in the INVITE (e.g. collect or toll-free call).

(5) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
Via: SIP/2.0/UDP Host(mta-o.provider)
```



```
Require: 100rel
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=0
Remote-Party-ID: John Smith <tel: +1-212-555-2222>
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 INVITE
Content-Disposition: precondition
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos: mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, Proxy-o authorizes the resources and forwards the following message to MTA-o. At this point Proxy-o has completed its transaction and does not maintain any more state for this call.

(6) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: Sip/2.0/UDP Host(mta-o.provider)
Require: 100rel
Remote-Party-ID: John Smith <tel: +1-212-555-2222>
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 INVITE
Content-Disposition: precondition
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
```



```
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos: mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, MTA-o sends the PRACK message directly to MTA-t using the IP address in the Contact header. MTA-o then reserves the resources needed, and sends a COMET message if successful. The COMET message, and the 200-OK messages in response to the PRACK and COMET are identical to that in the basic call flow (Section 10.1) and not shown here.

(7) PRACK:

```
PRACK sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
CSeq: 130 PRACK

v=0
```

```
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
a=qos: mandatory sendrecv
```

MTA-t reserves the resources as needed from the final SDP from the PRACK message. If successful, and upon receipt of the COMET message from MTA-o indicating it was successful, MTA-t changes the CODEC and sends the 200-OK final response.

(11) 200-OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider);branch=1"; via=Host(mta-o.provider)}K
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
CSeq: 129 INVITE
```


Upon receiving the 200-OK message, Proxy-t forwards the following 200-OK to Proxy-o, restoring the Via headers.

(12) 200-OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
Via: SIP/2.0/UDP Host(mta-o.provider)
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=0
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
```

```
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     CSeq: 129 INVITE
```

Upon receiving the 200-OK message, Proxy-o forwards the following 200-OK to MTA-o.

```
(13) 200-OK:
     SIP/2.0 200 OK
     Via: Sip/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     CSeq: 129 INVITE
```

Upon receiving the 200-OK message, MTA-o sends the following ACK message directly to MTA-t using the IP address in the Contact header of the previous 183 message. This completes the three-way handshake for the SIP INVITE exchange.

```
(14) ACK:
     ACK sip:Host(mta-t.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     CSeq: 129 ACK
```

10.17 Busy-Line-Verification

```
   PSTN G/W          CMS/MGC      Proxy-t2           MTA-t2

     |  (1) NTFY        |               |                   |
     |----------------->|               |                   |
     |  (2) CRCX        |               |                   |
     |<-----------------|               |                   |
     |  (3) ACK         |               |                   |
     |----------------->|               |                   |
     |                  |               |                   |
     |                  |  (4) INVITE   |                   |
     |                  |-------------->|   (5) INVITE      |
     |                  |               |------------------>|
     |                  |               |  (6) 183 SDP      |
     |                  |  (7) 183 SDP  |<------------------|
     |                  |<--------------|                   |
     |                  |  (8) PRACK    |                   |
     |                  |---------------------------------->|
     |                  |  (9)  200 OK  |                   |
     |                  |<----------------------------------|
     |                  |  (10) COMET   |                   |
     |                  |---------------------------------->|
     |                  |  (11) 200 OK  |                   |
     |                  |<----------------------------------|
     |                  |               | (14) 200 OK       |
     |                  |  (15) 200 OK  |<------------------|
     |                  |<--------------|                   |
     |                  |  (16) ACK     |                   |
     |                  |---------------------------------->|
     |                  |               |                   |
```

This service clearly requires the cooperation of the MTA. Further, there is a network database of phone numbers of customers who cannot be verified and/or broken into. It seems reasonable that a MTA that refuses to cooperate is merely one so marked in the current database.

Busy Line Verification sequence begins when the Operator at an OSPS console signals an E.164 number for verification over a special MF trunk group, to the PSTN gateway. The Media Gateway (MG) and Media Gateway Controller (MGC) recognize this special signaling, and generate an INVITE(BLV) to the number requested.

The normal call initiation sequence in TGCP is followed. The NTFY message (1) signals the MGC of a call request, and MGC uses the CRCX message (2) and ACK (3) to generate an appropriate SDP. That it is a OSPS trunk group is known to the MGC, which invokes the special header insertion.

MGC recognizes the trunk group as special BLV trunks, and generates a slightly modified INVITE message, by adding the Dcs-OSPS header.

(4) INVITE (BLV):

```
     INVITE sip:212-555-1111,lnp=212-237@dp-t.provider;user=phone
             SIP/2.0
     Via: SIP/2.0/UDP Host(mgc02.provider);branch=1
     Supported: 100rel, state
     Proxy-Require: dcs
     Remote-Party-ID: Operator <sip:Operator42@mgc02.provider>;
             rpi-type=operator
     Anonymity:  URL
     Dcs-OSPS: BLV
     Dcs-Gate: mgc02.provider/36123E5B
     Dcs-Billing-Info: Host(rks-o.provider)<OSPS/212-0/212-555-1111>
     Dcs-Billing-ID: Host(mgc02.provider):36123E5C:0152
     From: B64(SHA-1(0;time=36123E5B;seq=72))@localhost
     To: tel:555-1111
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     CSeq: 127 INVITE
     Contact: sip:mgc02.provider
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 mg101.provider
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     a=rtpmap:96 G726-32/8000
     m=audio 3380 RTP/AVP 0
     a=qos:mandatory sendrecv
     a=X-pc-codecs:96
```

Proxy-t authorizes the additional connection without regard to number of currently active connections, and passes the INVITE(BLV) to MTA-t.

(5) INVITE (BLV):

```
     INVITE sip:212-555-1111@mta-t.provider;user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t.provider);branch=1, {
     via="Host(mgc02.provider);branch=1"}K
```

```
Supported: 100rel, state
Require: state
```



```
Remote-Party-ID: Operator <sip:{type=remote-id;
        orig=sip:Operator42@mgc02.provider; anonymity=URL}K>;
        rpi-type=operator
Dcs-OSPS: BLV
Dcs-Gate: 44S10312
State: Host(dp-t.provider); state="{nexthop=sip:Host(mgc02.
        Provider); gate=Host(cmts-t.provider):4321/44S10312}K"
From: B64(SHA-1(0;time=36123E5B;seq=72))@localhost
To: tel:555-1111
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:mgc02.provider
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 mg101.provider
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3380 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

MTA-t does not respond to a BLV request with BUSY, nor does it perform call forwarding. The response is 183-Session-Progress, identical to that of a normal call given in Section 10.1, and completes identical to a normal call (but without the ringing phase).

MTA-t commits to the reserved resources, and begins to send voice packets to the Operator. The payload contains a copy of the packets generated at MTA-t.

If the designated line is onhook, MTA-t will generate silence packets and send to Operator. If the line is currently ringing or generating local ringback, MTA-t will generate a ringback tone pattern and sent to Operator.

The OSPS system scrambles the received voice packets, making the conversation unintelligible to the Operator. However, enough information is passed through the scrambled audio for the operator to accurately determine whether the line is in use, dialing, ringing, or idle.

A BLV call terminates when the OSPS signals a hangup over the MF trunk, resulting in a DCS BYE message. The MTA never terminates a BLV call.

10.18 Operator break-in

```
PSTN G/W          CMS/MGC      Proxy-t2             MTA-t2

 |  (1) NTFY         |               |                   |
 |------------------>|               |                   |
 |  (2) CRCX         |               |                   |
 |<------------------|               |                   |
 |  (3) ACK          |               |                   |
 |------------------>|               |                   |
 |                   |               |                   |
 |                   |  (4) INVITE   |                   |
 |                   |-------------->|   (5) INVITE      |
 |                   |               |------------------>|
 |                   |               |  (6) 183 SDP      |
 |                   |  (7) 183 SDP  |<------------------|
 |                   |<--------------|                   |
 |                   |  (8) PRACK    |                   |
 |                   |---------------------------------->|
 |                   |  (9)  200 OK  |                   |
 |                   |<----------------------------------|
 |                   |  (10) COMET   |                   |
 |                   |---------------------------------->|
 |                   |  (11) 200 OK  |                   |
 |                   |<----------------------------------|
 |                   |               | (14) 200 OK       |
 |                   |  (15) 200 OK  |<------------------|
```

```
  |                  |<--------------|                   |
  |                  |  (16) ACK     |                   |
  |                  |---------------------------------->|
  |  (17) NTFY       |               |                   |
  |----------------->|               |                   |
  |                  |  (18) INVITE  |                   |
  |                  |---------------------------------->|
  |                  |  (19) 200 OK  |                   |
  |                  |<----------------------------------|
  |                  |  (20) ACK     |                   |
  |                  |---------------------------------->|
  |                  |               |                   |
```

Emergency Interrupt is closely tied to BLV (previous appendix), since EI always begins with BLV. Messages (1) through (16) perform the BLV function, and are not repeated here.

At the end of the BLV call flow, instead of the OSPS releasing the trunk, OSPS generates an alerting tone. The Media Gateway (MG) detects activity on line and the Media Gateway Controller (MGC) generates INVITE(EI) and sends it direct to MTA-t.

```
(18) INVITE(EI):
     INVITE sip:Host(mta-t.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mgc02.provider)
     From: B64(SHA-1(0;time=36123E5B;seq=72))@localhost
     To: tel:555-1111
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     CSeq: 130 INVITE
     Dcs-OSPS: EI
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 mgc02.provider
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     m=audio 3380 RTP/AVP 0
```

MTA verifies that BLV is already active, and that the EI request matches From, To, Call-ID. If so, it responds with 200-OK. SDP is not needed unless there is a change in the session description from that of the BLV.

(19) 200 OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(mgc02.provider)
From: B64(SHA-1(0;time=36123E5B;seq=72))@localhost
To: tel:555-1111
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
CSeq: 130 INVITE
```

MGC responds with the standard SIP ACK message.

MTA has several choices of how to do the EI function. 1) put current call on hold and connect user to operator. 2) perform local mixing of current call and stream from OSPS, so both participants in existing call hear and can talk to the operator. 3) allocate a bridge and treat this just like a 3-way call.

10.19 Lawfully Authorized Electronic Surveillance

Calls from and to surveillance subjects behave just like a normal call flows. Additional messages will be sent from the CMS to the Electronic Surveillance Delivery Function telling the known signaling information. One additional parameter is sent to the CMTS in the Gate-Set command, telling it to copy all the voice data packets and send the copy to the Law Enforcement Access Point. There is no change to the MTA-CMS message exchanges due to electronic surveillance. This is an absolute requirement due to the non-intrusive requirement of the electronic surveillance statute.

In most cases, there is no change to the CMS-CMS message exchanges due to electronic surveillance. This basic design keeps the knowledge of surveillance as localized as possible. If a call originator is under surveillance, the surveillance is done at the originating CMTS; the call destination does not know in any way that it is happening. If a call destination is under surveillance, the surveillance is done at the terminating CMTS; the call originator does not know in any way that it is happening. If both the call originator and call destination are under surveillance, the interception is done twice. So it goes.

The only situation where CMS-CMS message exchanges are extended to support electronic surveillance is in cases of call redirection.

When a subject under surveillance initiates a call transfer, it is required that the new call also be intercepted. Therefore the notification of surveillance is passed in that CMS-CMS INVITE message. Further, when the new destination is unable to perform the required interception (e.g. redirection to a network server such as voicemail), the 183 response contains additional information telling the originator to perform the surveillance. These situations are shown in the following examples.

10.19.1 Call-Forwarding-Unconditional with Forwarder under Surveillance

```
MTA-o              Proxy-o      Proxy-t                MTA-t

 |     INVITE       |               |                   |
 |----------------->|    INVITE     |                   |
 |                  |-------------->|   (1)INVITE       |
```

```
    |                  |               |------------------>|
    |                  |               |  (2)302 Redirect  |
    |                  |               |<------------------|
    |                  |   (3)302      |                   |
    |                  |<--------------|   (4)ACK          |
    |                  |               |------------------>|
    |                  |   (5)ACK      |                   |
    |                  |-------------->|                   |
    |                  |
    |                  |             Proxy-f             MTA-f
    |                  |               |                   |
    |                  |  (6)INVITE    |                   |
    |                  |-------------->|   (7)INVITE       |
    |                  |               |------------------>|
    |                  |               |  (8)183 SDP       |
    |                  |               |<------------------|
    |                  |               |                   |
```

For this example, consider a call from MTA-o to MTA-t (with MTA-t under surveillance). MTA-t has established call forwarding-unconditional. The basic call flow for call-forwarding is identical to that given in Section 10.5, and only the differences are noted here.

```
(1) INVITE:
     INVITE sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"; via=Host(mta-o.provider)}K
     Supported: 100rel, state
     Require: state
     Remote-Party-ID: John Doe <tel:+1-212-555-1111>
     Media-Authorization: 31S14621
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider); gate=Host(cmts-t.provider):4321/31S14621;
             state="Host(dp-o.provider); nexthop=sip:555-
             1111@Host(mta-o.provider); gate=Host(cmts-
```



```
             o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
             2222; num-redirects=0"}K"
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
     Contact: sip:Host(mta-o.provider)
```

```
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this message, MTA-t determines that the line associated with 212-555-2222 is having all calls forwarded. It issues a REDIRECT (302) response to indicate that it wants the call forwarded. This message carries the forwarding number in the Contact header.

(2) 302-Redirect

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
Remote-Party-ID: John Smith <tel:555-2222>
Anonymity: off
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: tel:555-3333
```

Proxy-t knows that MTA-t is under surveillance, and includes the Dcs-Laes header in the 302-Redirect response sent back to CMS-o.

This header contains the delivery function information needed by the new destination of the call.

(3) 302-Redirect

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP Host(dp-o.provider); branch = 1
Via: SIP/2.0/UDP Host(mta-o.provider)
Proxy-Require: dcs
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
        555-1111/212-555-2222>
Dcs-Billing-Info: Host(rks-t.provider)<4278-9865-8765-9000/212-
        555-2222/212-555-3333>
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=0
Remote-Party-ID: John Smith <tel:+1-212-555-2222>
Anonymity: off
Dcs-Laes: Host(df-t)/Host(df-t);surveillancekey
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: tel:+1-212-555-3333
```

Proxy-o determines the Proxy-f for the E.164 number 212-555-3333 when it receives the 302-Redirect message. It generates an INVITE message and sends it to Proxy-f. Proxy-o adds the Dcs-Laes header to this INVITE, with the delivery function information received from Proxy-t. Proxy-o adds the Dcs-Redirect header giving the information about this call redirection.

(6) INVITE:

```
INVITE sip:+1-212-555-3333;rn=+1-212-265-3333;
        npdi=yes@Host(dp-f) ;user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-o.provider); branch = 2
Via: SIP/2.0/UDP Host(mta-o.provider);
Supported: 100rel, state
Require: state
Proxy-Require: dcs, state
Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
Anonymity: Off
Dcs-Gate: Host(cmts-o.provider):3612/17S30124/37FA1948 required
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
        555-1111/212-555-2222>
Dcs-Billing-Info: Host(rks-t.provider)<4278-9865-8765-9000/212-
        555-2222/212-555-3333>
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=1
```

```
Dcs-Laes: Host(df-t)/Host(df-t);surveillancekey
Dcs-Redirect: <tel:+1-212-555-2222> <tel:+1-212-555-2222> 1
```



```
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
CSeq: 127 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE, Proxy-f queries the directory server to determine the IP address (MTA-f) associated with 212-555-3333. It then forwards the INVITE message to MTA-f. Included in the State header is the additional surveillance information.

(7) INVITE:

```
INVITE sip:555-3333@Host(mta-f.provider); user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-f.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
Supported: 100rel, state
Require: state
Remote-Party-ID: John Doe <tel:+1-212-555-1111>
Media-Authorization: 22S21718
State: Host(dp-f.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-f.provider):4321/22S21718;
        laes=full; redirect=<tel:+1-212-555-2222> <tel:+1-212-
        555-2222> 1;  state="Host(dp-o.provider);
        nexthop=sip:555-1111@Host(mta-o.provider);
        gate=Host(cmts-o.provider):3612/17S30124; orig-
```

```
            dest=tel:+1-212-555-2222; num-redirects=1"}K"
    From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
            seq=72))@localhost>
    To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
    Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
    Cseq: 127 INVITE
    Contact: sip:Host(mta-o.provider)
    Content-Type: application/sdp
    Content-length: (.)
```



```
    v=0
    o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
    s=-
    c= IN IP4 Host(mta-o.provider)
    b=AS:64
    t=907165275 0
    a=X-pc-csuites:312F
    a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
    a=rtpmap:0 PCMU/8000
    a=rtpmap:96 G726-32/8000
    m=audio 3456 RTP/AVP 0
    a=qos:mandatory sendrecv
    a=X-pc-codecs:96
```

The subsequent signaling call flows are identical to those shown in Section 10.5.

10.19.2 Call-Transfer-Blind with Transferer under Surveillance

```
   MTA-o              Proxy-o      Proxy-t              MTA-t

     |                  |               |                   |
     |<-----------------call-in-progress------------------->|
     |                  |               |  (1) REFER        |
     |                  |  (2) REFER    |<------------------|
     |   (3) REFER      |<--------------|                   |
     |<-----------------|               |                   |
     |                  |
     |                  |          Proxy-f              MTA-f
     |  (4) INVITE      |               |                   |
     |----------------->|  (5) INVITE   |                   |
     |                  |-------------->|  (6) INVITE       |
     |                  |               |------------------>|
     |                  |               |  (7) 183 SDP      |
```

```
|                  | (8) 183 SDP   |<------------------|
|    (9) 183 SDP   |<--------------|                   :
|<-----------------|               :                   :
:                  :               :   (10) 200 OK     |
:                  :  (11) 200 OK  |<------------------|
|    (12) 200 OK   |<--------------|                   |
|<-----------------|               |                   |
|                  |
|                  |          Proxy-t              MTA-t
|  (13) 200 OK     |               |                   |
|----------------->| (14) 200 OK   |                   |
|                  |-------------->|  (15) 200 OK      |
|                  |   (16) ACK    |------------------>|
|<-----------------------------------------------------|
|                  |   (17) BYE    |                   |
|----------------------------------------------------->|
|                  |  (18) 200 OK  |                   |
```



```
|<-----------------------------------------------------|
|                  |
```

For this example, consider an existing call initiated by MTA-o to MTA-t (with MTA-t under surveillance), with the following call identification. The only difference from a normal call from MTA-o to MTA-t, as given elsewhere in this specification, is the addition of "laess=full" in the encrypted call state information associated with Proxy-t.

```
MTA-t state for call from MTA-o to MTA-t
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Contact: sip:Host(mta-o.provider)
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider); gate=Host(cmts-t.provider):4321/31S14621;
             laes=full; state="Host(dp-o.provider); nexthop=sip:555-
             1111@Host(mta-o.provider); gate=Host(cmts-
             o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
             2222; num-redirects=0"}K"
     Dcs-Billing-Info: Host(rks-o.provider)/04FA37<5123-0123-4567-
             8900/212-555-1111/212-555-2222>
     Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
```

The basic call flow for call transfer is identical to that given in Section 10.11, and only the differences are noted here.

MTA-t desires to transfer the existing call to 555-3333 and issues a REFER message, identical to (1) in Section 10.11. Proxy-t decrypts the State information and sees the "laes=full." It inserts one additional header component into the Refer-to header, giving the local Electronic Surveillance Delivery Function's (DF's) address information and security key to use for messages to the DF. Since surveillance was marked as "full," it adds the DF's address for both signaling information and for call content.

```
(2) REFER:
     REFER sip: Host(dp-o.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t.provider)
     Via: SIP/2.0/UDP Host(mta-t.provider)
     Supported: 100rel, state
     Proxy-Require: dcs
     State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
             o.provider); gate=Host(cmts-o.provider):3612/17S30124;
             orig-dest=tel:+1-212-555-2222; num-redirects=0
     Refer-to: tel:+1-212-555-3333? Dcs-Billing-Info= Host(rks-
             o.provider)<5123-0123-4567-8900/212-555-1111/212-555-
             2222> & Dcs-Billing-Info= Host(rks-t.provider)<4278-
             9865-8765-9000/212-555-2222/212-555-3333> & Dcs-
             Billing-ID= Host(dp-o.provider): 36123E5C:0152 & Dcs-
             Laes= Host(df-t)/Host(df-t);surveillancekey & Dcs-
             Redirect=<tel:+1-212-555-2222><tel:+1-212-555-2222>1
```



```
     Remote-Party-ID: John Smith <tel:+1-212-555-2222>
     From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 8001 REFER
     Referred-by: sip:B64(SHA-1(555-2222; time=36123E5B;
             seq=73))@localhost
```

Proxy-o includes the Dcs-Laes information in the encrypted URL given to MTA-t.

```
(3) REFER:
     REFER sip: Host(mta-o.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(dp-o.provider), {via="Host(dp-
             t.provider); branch=1"; via=Host(mta-t.provider)}K
     Supported: 100rel, stateRefer-to:  sip:{type=transfer;
     dest=tel:+1-212-555-3333;       billing-id=Host(dp-o.provider):
     36123E5C:0152;  expires=<timestamp>; billing-info=Host(rks-
             o.provider)<5123-0123-4567-8900/212-555-1111/212-555-
```

```
        2222>; billing-info=Host(rks-t.provider)<4278-9865-
        8765-9000/212-555-2222/212-555-3333>; laes="Host(df-
        t)/Host(df-t);surveillancekey"; redirect=<tel:+1-212-
        555-2222><tel:+1-212-555-2222>1}K@Host(dp-
        o.provider); user=private
From: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
To: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 8001 REFER
Referred-by: sip:B64(SHA-1(555-2222; time=36123E5B;
        seq=73))@localhost
```

After processing the REFER, MTA-o issues a INVITE to MTA-f. In addition to the standard headers carried in an INVITE message, the encrypted Dcs-Laes fields received in the REFER message are copied into the INVITE message.

(4) INVITE:

```
INVITE sip:{type=transfer; dest=tel:+1-212-555-3333; billing-
        id=Host(dp-o.provider): 36123E5C:0152;
        expires=<timestamp>; billing-info=Host(rks-
        o.provider)<5123-0123-4567-8900/212-555-1111/212-555-
        2222>; billing-info=Host(rks-t.provider)<4278-9865-
        8765-9000/212-555-2222/212-555-3333>; laes="Host(df-
        t)/Host(df-t);surveillancekey"; redirect=<tel:+1-212-
        555-2222><tel:+1-212-555-2222>1}K@Host(dp-
        o.provider); user=private SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
Supported: 100rel, state
Remote-Party-ID: John Doe <tel:555-1111>
Anonymity: Off
```



```
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E98;
        seq=74))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E98; seq=75))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E98;seq=74))@localhost
Cseq: 129 INVITE
Contact: sip:Host(mta-o.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
```

```
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

When the Proxy-o receives the INVITE it decrypts the header information to find the real destination, and discovers the Dcs-Laes information. This is included in the INVITE message sent to Proxy-f. Proxy-o also includes a Dcs-Redirect header giving the original destination for the call from MTA-o, the forwarding endpoint, and the number of redirections that have occurred to this call.

(5) INVITE:

```
INVITE sip: +1-212-555-3333;rn=+1-212-265-3333;
        npdi=yes@Host(dp-f);user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-o.provider); branch=1;
Via: SIP/2.0/UDP Host(mta-o.provider);
Supported: 100rel, state
Require: state
Proxy-Require: dcs, state
Remote-Party-ID: John Doe <tel:+1-212-555-1111>
Anonymity: Off
Dcs-Gate: Host(cmts-o.provider):3612/17S30124/37FA1948
Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
        555-1111/212-555-2222>
Dcs-Billing-Info: Host(rks-t.provider)<4278-9865-8765-9000/212-
        555-2222/212-555-3333>
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124;
        orig-dest=tel:+1-212-555-2222; num-redirects=1
Dcs-Laes: Host(df-t)/Host(df-t);surveillancekey
Dcs-Redirect: <tel:+1-212-555-2222> <tel:+1-212-555-2222> 1
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E98;
        seq=74))@localhost>
```



```
To: sip:B64(SHA-1(555-2222; time=36123E98; seq=75))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E98;seq=74))@localhost
Cseq: 129 INVITE
Contact: sip:Host(mta-o.provider)
```

```
      Content-Type: application/sdp
      Content-length: (.)

      v=0
      o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
      s=-
      c= IN IP4 Host(mta-o.provider)
      b=AS:64
      t=907165275 0
      a=X-pc-csuites:312F
      a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
      a=rtpmap:0 PCMU/8000
      a=rtpmap:96 G726-32/8000
      m=audio 3456 RTP/AVP 0
      a=qos:mandatory sendrecv
      a=X-pc-codecs:96
```

Upon receiving this INVITE, Proxy-f includes the surveillance information in its Gate-Set command, and passes the call-identifying information to its local Electronic Surveillance Delivery Function (DF-f) who passes the information on to DF-t. The encrypted State value stored at MTA-f includes the surveillance parameters needed for possible mid-call transfers.

(6) INVITE:

```
      INVITE sip:555-3333@Host(mta-f.provider); user=phone SIP/2.0
      Via: SIP/2.0/UDP Host(dp-f.provider), {via="Host(dp-
              o.provider); branch=1"; via=Host(mta-o.provider)}K
      Supported: 100rel, state
      Require: state
      Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
      Media-Authorization: 31S14621
      State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
              o.provider); gate=Host(cmts-t.provider):4321/31S14621;
              laes=full; redirect=<tel:+1-212-555-2222><tel:+1-212-
              555-2222>1; state="Host(dp-o.provider);
              nexthop=sip:555-1111@Host(mta-o.provider);
              gate=Host(cmts-o.provider):3612/17S30124; orig-
              dest=tel:+1-212-555-2222; num-redirects=0"}K"
      From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E98;
              seq=74))@localhost>
      To: sip:B64(SHA-1(555-2222; time=36123E98; seq=75))@localhost
      Call-ID: B64(SHA-1(555-1111;time=36123E98;seq=74))@localhost
      Cseq: 129 INVITE
      Contact: sip:Host(mta-o.provider)
      Content-Type: application/sdp
      Content-length: (.)

      v=0
```

```
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

When the new call completes, MTA-o acknowledges receipt of the REFER by sending a 200-OK to MTA-t. This is identical to Section 10.11.

Remainder of the call is identical to the basic call flow shown in Section 10.11, and is not repeated here.

10.19.3 Call-Transfer with new destination unable to do interception

If CMS-f determines that it is unable to perform the required surveillance, it passes the request back to CMS-o. This occurs, for instance, if the new destination of this redirected call is a voicemail server, or a announcement server, or a bridge server, or any other network server that does not implement the capability to intercept the media packets. CMS-f includes the Dcs-Laes header in the 183-Session-Progress message as follows:

```
(8) 183-Session-Progress:
     SIP/2.0 183 Session Progress
     Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Require: 100rel
     Proxy-Require: dcs
     State: Host(dp-f.provider); nexthop=sip:555-3333Host(mta-
             f.provider); gate=Host(cmts-f.provider):4321/31S14621;
             orig-dest=tel:+1-212-555-1111; num-redirects=0
     State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
             o.provider); gate=Host(cmts-o.provider):3612/17S30124;
             orig-dest=tel:+1-212-555-2222; num-redirects=1
     Dcs-Gate: Host(cmts-t.provider):4321/31S14621/37FA1948
     Remote-Party-ID: John Smith <tel:+1-212-555-2222>
     Anonymity: off
     Dcs-Laes: Host(df-o)/Host(df-o);surveillancekey
```

```
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
```



```
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

Proxy-o performs the surveillance in its Gate-Set command to its CMTS. Remainder of the call flow is identical to a normal call.

10.19.4 Call-Transfer-Consultative with Transferer under Surveillance

```
MTA-o    MTA-t1    Proxy-o      Proxy-t2             MTA-t2

 |  (1) REFER        |               |                   |
 |------------------>|  (2) REFER    |                   |
 |        |          |-------------->|   (3) REFER       |
 |        |          |               |------------------>|
 |        |                          |                   |
 |        |      Proxy-t1            | (4) INVITE        |
 |        |          |  (5)INVITE    |<------------------|
 |        |(6)INVITE|<---------------|                   |
 |        |<--------|                |                   |
 |        | (7)SDP  |                |                   |
 |        |-------->|  (8) SDP       |                   |
 |        |         |--------------->|    (9) SDP        |
 |        |         |  (10) PRACK    |------------------>|
 |        |<-------------------------------------------- |
 |        |         |  (11) 200 OK   |                   |
 |        |--------------------------------------------->|
```

```
    |        |         |  (12) COMET   |                   |
    |        |<--------------------------------------------|
    |        |         |  (13) 200 OK  |                   |
    |        |-------------------------------------------->|
    |        |(14)200  |               |                   |
    |        |-------->| (15) 200 OK   |                   |
    |        |         |-------------->|   (16) 200 OK     |
    |        |         |  (17) ACK     |------------------>|
    |        |<--------------------------------------------|
    |        |         |               |                   |
    |        |      Proxy-o            | (17) 200 OK       |
    |        |         |  (18)200 OK   |<------------------|
```



```
    |    (19) 200 OK   |<--------------|                   |
    |<-----------------|               |                   |
    |        |         |  (20) BYE     |                   |
    |----------------------------------------------------->|
    |        |         |  (21) 200 OK  |                   |
    |<-----------------------------------------------------|
    |(22)BYE |         |               |                   |
    |------->|         |               |                   |
    |(23)200 |         |               |                   |
    |<-------|         |               |                   |
```

After some period of consultation, MTA-o initiates a transfer of the call from MTA-t1 to the new destination, MTA-t2. This involves placing the second call on hold (message sequence described earlier), and sending an INVITE(also,replace) message to MTA-t2, giving it the information about the call with MTA-t1 in the Also: header. The INVITE message, since it changes parties involved in the call, is routed through the proxies. The sequence is shown in the figure above, and detailed below.

For this example, consider a call initiated by MTA-t1 to MTA-x, with MTA-x under surveillance, where MTA-x performed a blind transfer to MTA-o. The call between MTA-t1 and MTA-o is therefore under surveillance. MTA-o desires to transfer the call (with consultation) to MTA-t2. After placing the call to MTA-t2, and placing that call on hold, MTA-o initiates a transfer by sending a REFER to MTA-t2, routed through the proxies.

The basic call flow for consultative transfer is identical to that given in Section 10.12, and only the differences due to surveillance are noted here.

```
   (1) REFER:
        REFER sip: Host(mta-t2.provider) SIP/2.0
        Via: SIP/2.0/UDP Host(mta-o.provider)
        Supported: 100rel, state
        Remote-Party-ID: John Doe <tel:555-1111>
        Anonymity: off
        Refer-to: tel:+1-212-555-2222 ? Call-ID=B64(SHA-1(555-
                1111;time=36124033;seq=72) & Referred-by=tel:555-1111
                & State= Host(dp-o.provider);
                state="{nexthop=sip:Host(dp-t1.provider);
                gate=Host(cmts-o.provider):3612/17S30124; laes=full;
                redirect=<tel:+1-212-555-7777><tel:+1-212-555-1111>2;
                state="Host(dp-t1.provider); nexthop=sip:555-
                2222@Host(mta-t1.provider); gate=Host(cmts-
                t1.provider):4321/31S14621; orig-dest=tel:+1-212-555-
                1111; num-redirects=0"}K"
        State: Host(dp-o.provider); state="{gate= Host(cmts-
                o.provider): 3612/3S10782, nexthop=sip:+1-212-555-
                3333;rn=+1-212-256-3333@Host(dp-t2.provider),
                state="Host(dp- t2.provider); nexthop=sip:555-
```



```
                3333@Host(mta-  t2.provider); gate=Host(cmts-
                t2.provider):4321/31S14621; orig-dest=tel:+1-212-555-
                1111; num-redirects=0"}K"
        From: sip:B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
        To: tel:555-3333
        Call-ID: B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
        CSeq: 133 REFER
        Referred-by: sip:B64(SHA-1(555-1111;time=36124125;
                seq=23))@localhost
```

Proxy-o decrypts the State information in the Refer-To header to determine the local state information. Proxy-o inserts billing information and Electronis Surveillance information into the Refer-To header. Proxy-o then forwards the message to Proxy-t1.

```
   (2) REFER:
        REFER sip: Host(dp-o.provider) SIP/2.0
        Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
        Via: SIP/2.0/UDP Host(mta-o.provider)
        Supported: 100rel, state
        Proxy-Require: dcs
        State: Host(dp-t2.provider); nexthop=sip:555-3333@Host(mta-
                t2.provider); gate=Host(cmts-
                t2.provider):4321/31S14621; orig-dest=tel:+1-212-555-
                1111; num-redirects=0
```

```
     Refer-to: tel:+1-212-555-2222? Call-ID=B64(SHA-1(555-
             1111;time=36124033;seq=72) & Referred-by=tel:555-1111
             & Dcs-Billing-Info= Host(rks-t1.provider)<4278-9865-
             8765-9000/212-555-2222/212-555-1111> & Dcs-Billing-
             Info= Host(rks-t2.provider)<5123-0123-4567-8900/212-
             555-1111/212-555-3333> & Dcs-Billing-ID= Host(dp-
             o.provider): 36123E5C:0152 & Dcs-Laes=Host(df-
             o)/Host(df-o);securitykey & Dcs-Redirect=<tel:+1-212-
             555-7777><tel:+1-212-555-1111>2
     Remote-Party-ID: John Smith <tel:+1-212-555-2222>
     From: sip:B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     To: tel:555-3333
     Call-ID: B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     CSeq: 133 REFER
     Referred-by: sip:B64(SHA-1(555-1111;time=36124125;
             seq=23))@localhost
```

Proxy-t2 forwards the REFER message to MTA-t2 after encrypting the destination of the transfer, and the Electronic Surveillance headers.

(3) REFER:

```
     REFER sip: 555-3333@Host(mta-t2.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(dp-t2.provider), {via="Host(dp-
             o.provider); branch=1"; via=Host(mta-o.provider)}K
     Supported: 100rel, state
     Refer-to:  sip:{type=transfer; dest=tel:+1-212-555-2222;
             billing-id=Host(dp-o.provider): 36123E5C:0152;
```



```
             expires=<timestamp>; billing-info= Host(rks-
             t1.provider)<4278-9865-8765-9000/212-555-2222/212-555-
             1111> ; billing-info= Host(rks-t2.provider)<5123-0123-
             4567-8900/212-555-1111/212-555-3333>; laes=Host(df-
             o)/Host(df-o);securitykey & redirect=<tel:+1-212-555-
             7777><tel:+1-212-555-1111>2}K@Host(dp-t2.provider);
             user=private ? Call-ID=B64(SHA-1(555-
             1111;time=36124033;seq=72) & Referred-by=tel:555-1111
     Remote-Party-ID: John Smith <tel:+1-212-555-2222>
     From: sip:B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     To: tel:555-3333
     Call-ID: B64(SHA-1(555-1111;time=36124125;seq=23))@localhost
     CSeq: 133 INVITE
     Referred-by: sip:B64(SHA-1(555-1111;time=36124125;
             seq=23))@localhost
```

After processing the REFER, MTA-t2 issues a INVITE to MTA-t1. In addition to the standard headers carried in an INVITE message, the encrypted {Dcs-Laes, Dcs-Redirect} fields received in the REFER message are copied into the Request-URI of the INVITE message. These fields indicate the surveillance information.

(4) INVITE:

```
INVITE sip:{type=transfer; dest=tel:+1-212-555-2222; billing-
        id=Host(dp-o.provider): 36123E5C:0152;
        expires=<timestamp>; billing-info= Host(rks-
        t1.provider)<4278-9865-8765-9000/212-555-2222/212-555-
        1111> ; billing-info= Host(rks-t2.provider)<5123-0123-
        4567-8900/212-555-1111/212-555-3333>; laes=Host(df-
        o)/Host(df-o);securitykey & redirect=<tel:+1-212-555-
        7777><tel:+1-212-555-1111>2}K@Host(dp-t2.provider);
        user=private SIP/2.0
Via: SIP/2.0/UDP Host(mta-t2.provider)
Supported: 100rel, state
Remote-Party-ID: John Smith <tel:555-3333>
Anonymity: Off
From: "Alien Blaster" <sip:B64(SHA-1(555-3333; time=36124172;
        seq=74))@localhost>
To: sip:B64(SHA-1(555-3333; time=36124172; seq=75))@localhost
Call-ID: B64(SHA-1(555-1111;time=36124033;seq=72)@localhost
Cseq: 129 INVITE
Referred-by: tel:555-1111
Contact: sip:Host(mta-t2.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-t2.provider)
b=AS:64
t=907165275 0
```



```
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

When the Proxy-t2 receives the INVITE it first decrypts the header information to find the real destination for the call, and the surveillance information.

(5) INVITE:

```
INVITE sip: +1-212-555-2222;rn=+1-212-265-2222;
        npdi=yes@Host(dp-t1);user=phone SIP/2.0
Via: SIP/2.0/UDP Host(dp-t2.provider); branch=1;
Via: SIP/2.0/UDP Host(mta-t2.provider);
Supported: 100rel, state
Require: state
Proxy-Require: dcs, state
Remote-Party-ID: John Smith <tel:+1-212-555-3333>
Anonymity: Off
Dcs-Gate: Host(cmts-t2.provider):3612/17S30124/37FA1948
Dcs-Billing-Info: Host(rks-t1.provider)<4278-9865-8765-
        9000/212-555-2222/212-555-1111>
Dcs-Billing-Info: Host(rks-t2.provider)<5123-0123-4567-
        8900/212-555-1111/212-555-3333>
Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
Dcs-Laes: Host(df-o)/Host(df-o);securitykey
Dcs-Redirect: <tel:+1-212-555-7777><tel:+1-212-555-1111>2
From: "Alien Blaster" <sip:B64(SHA-1(555-3333; time=36124172;
        seq=74))@localhost>
To: sip:B64(SHA-1(555-3333; time=36124172; seq=75))@localhost
Call-ID: B64(SHA-1(555-1111;time=36124033;seq=72)@localhost
Cseq: 129 INVITE
Referred-by: tel:555-1111
Contact: sip:Host(mta-t2.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio 3456 RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE, Proxy-t1 queries the directory server to determine the IP address (MTA-t1) associated with 212-555-2222. It then forwards the INVITE message to MTA-t1, after stripping off all of the billing fields, and adding the encrypted state information. Remainder of the call completes as shown in Section 10.12.

10.20 Privacy with Application-level Anonymizer

```
   MTA-o     ANON-o     Proxy-o     Proxy-t    ANON-t     MTA-t

    |   (1)Invite         |          |          |          |
    |-------------------->|          |          |          |
    |          | (2)CREAT |          |          |          |
    |          |<---------|(4)Invite | (5)CREAT |          |
    |          |-(3)ACK-->|--------->|--------->|          |
    |          |          |          |<-(6)ACK--|          |
    |          |          |          |(7)Invite |          |
    |          |          |          |-------------------->|
    |          |          |          |   (8) 183 SDP       |
    |          |          |          |<--------------------|
    |          |          |          |(9)MODIFY |          |
    |          |          |          |--------->|          |
    |          |          |(11)183SDP|<-(10)ACK-|          |
    |          |(12)MODIFY|<---------|          |          |
    |          |<---------|          |          |          |
    |(14)183SDP|-(13)ACK->|          |          |          |
    |<--------------------|          |          |          |
    |(15)PRACK |           (16) PRACK           |(17)PRACK |
    |--------->|------------------------------->|--------->|
    |(20)200 OK|           (19) 200 OK          |(18)200 OK|
    |<---------|<-------------------------------|<---------|
    |(21)COMET |           (22) COMET           |(23)COMET |
    |--------->|--------- --------------------->|--------->|
    |(26)200 OK|           (25) 200 OK          |(24)200 OK|
    |<---------|----------|----------|----------|----------|
    |          |          |          |          |(27)180 Rg|
    |          |          |(28) 180  |<--------------------|
    |       (29)180 Ring  |<---------|          |          |
    |<---------|----------|          |          |          |
    |(30)PRACK |           (31) PRACK           |(32)PRACK |
    |--------->|------------------------------->|--------->|
    |(35)200 OK|           (34) 200 OK          |(33)200 OK|
    |<---------|<-------------------------------|----------|
    |          |          |          |          |          |
    |          |          |          |      (36) 200 OK    |
    |          |          |(37)200 OK|<--------------------|
    |      (38) 200 OK    |<---------|          |          |
    |<--------------------|          |          |          |
    |(39) ACK  |          | (40) ACK |          |(41) ACK  |
    |--------->|--------------------------------|--------->|
```

```
     |          |          |          |          |          |
```



```
     |<--------------------Active  Call-------------------->|
     |(42) BYE  |          |(43) BYE  |          |(44) BYE  |
     |--------->|------------------------------->|--------->|
     |(47)200 OK|          |(46)200 OK|          |(45)200 OK|
     |<---------|<-------------------------------|<---------|
```

The call begins identically to that shown in Section 10.1 of a basic MTA-originated call. The only difference at the originating MTA is that the Anonymity header is set to "Full" or includes "IPAddr."

(1) INVITE:

```
     INVITE sip:555-2222@Host(DP-o);user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Supported: 100rel, state
     Proxy-Require: privacy
     Remote-Party-ID: John Doe <tel:555-1111>
     Anonymity: Full
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
     Contact: sip:Host(mta-o.provider)
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mta-o.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuite:312F
     a=rtpmap:0 PCMU/8000
     a=rtpmap:96 G726-32/8000
     m=audio 3456 RTP/AVP 0
     a=qos:mandatory sendrecv
     a=X-pc- codecs:96
```

In addition to its normal functions, Proxy-o also checks the Anonymity indication which is set to "full", so both caller-id/calling name blocking and IP address privacy must be provided. Proxy-o therefore contacts the anonymizer to create an anonymous session:

```
(2) ANON_Create:
     ANON_Create
     Endpoint1:      Host(mta-o.provider):Port(mta-o.provider)
     Endpoint2:
```

The anonymizer responds back with an anonymizer endpoint address:

```
(3) ANON_Ack:
```



```
     ANON_Ack
     AnonAddr:       Host(ann-o.provider):Port(ann-o.provider)
```

The anonymizer implements a packet relay and call signaling gateway between the two endpoints. The first endpoint specified in the ANON_Create will receive the anonymizer service. Any packet received for the anonymizer address specified will be forwarded to Endpoint1. Any packet sent by Endpoint1 to the anonymizer address will be forwarded to Endpoint2, but now with a source address of AnonAddr.

Having received the anonymizer address for the call, Proxy-o generates the following INVITE message and sends it to Proxy-t. Proxy-omodifies a number of parameters to the INVITE message.

```
(4) INVITE:
     INVITE sip:+1-212-555-2222;rn=+1-212-234-2222;
             npdi=yes@Host(dp-t);user=phone SIP/2.0
     Via: SIP/2.0/UDP Host(DP-o.provider);branch=1
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Supported: 100rel, state
     Require: state
     Proxy-Require: dcs, state, privacy
     Remote-Party-ID: John Doe; <tel:+1-212-555-1111>
     Anonymity: Full
     Dcs-Gate: Host(cmts-o.provider):3612/17S30124/37FA1948 required
     Dcs-Billing-Info: Host(rks-o.provider)<5123-0123-4567-8900/212-
             555-1111/212-555-2222>
     State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
             o.provider); gate=Host(cmts-o.provider):3612/17S30124;
             orig-dest=tel:+1-212-555-2222; num-redirects=0
     Dcs-Billing-ID: Host(dp-o.provider):36123E5C:0152
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
     Contact: sip:Host(ann-o.provider)
```

```
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(ann-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio Port(ann-o) RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```


Proxy-t also checks the Anonymity indication which is set to "full", so both caller-id/calling name blocking and IP address privacy must be provided. We furthermore assume, that MTA-t has requested privacy, i.e. the originating party must not be able to tell the IP-address of MTA-t. Proxy-t therefore contacts an anonymizer to create an anonymous session:

```
(5) ANON_Create:
     ANON_Create
     Endpoint1:
     Endpoint2:      Host(ann-o.provider):Port(ann-o.provider)
```

The anonymizer responds back with an anonymizer endpoint address:

```
(6) ANON_Ack:
     ANON_Ack
     AnonAddr:       Host(ann-t.provider):Port(ann-t.provider)
```

The anonymizer implements a packet relay and call signaling gateway between the two endpoints. The first endpoint specified in the ANON_Create will receive the anonymizer service. Any packet received for the anonymizer address specified will be forwarded to Endpoint1. Any packet sent by Endpoint1 to the anonymizer address will be forwarded to Endpoint2, but now with a source address of AnonAddr. Proxy-t now generates the following INVITE message and sends it to MTA-t.

```
(7) INVITE:
     INVITE sip:555-2222@Host(mta-t.provider); user=phone SIP/2.0
```

```
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
Supported: 100rel, state
Require: state
Remote-Party-ID: <sip:{type=remote-id; orig=tel:+1-212-555-
        1111; anonymity=full}K@Host(dp-t.provider);
        user=private>;  rpi-id=private
Media-Authorization: 31S14621
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Contact: sip:Host(ann-t.provider):Port(ann-t.provider)
Content-Type: application/sdp
Content-length: (.)
```



```
v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(ann-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
a=rtpmap:96 G726-32/8000
m=audio Port(ann-t.provider) RTP/AVP 0
a=qos:mandatory sendrecv
a=X-pc-codecs:96
```

Upon receiving this INVITE, MTA-t authenticates that the message came from Proxy-t using IPSec. MTA-t checks the telephone line associated with the E.164-t to see if it is available. If it is available, MTA-t looks at the capability parameters in the Session Description Protocol (SDP) part of the message and determines which media channel parameters it can accommodate for this call. MTA-t stores the INVITE message, including the encrypted State parameters, for later use.   MTA-t puts this line in the "busy" state (so any other call attempts are rejected until this call clears), generates the following 183-Session-Progress response, and sends it to Proxy-t. MTA-t starts timer (T-proxy-response).

(8) 183-Session-Progress:

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
        o.provider); branch=1"; via=Host(mta-o.provider)}K
Require: 100rel
State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
        o.provider); gate=Host(cmts-t.provider):4321/31S14621;
        state="Host(dp-o.provider); nexthop=sip:555-
        1111@Host(mta-o.provider); gate=Host(cmts-
        o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
        2222; num-redirects=0"}K"
Remote-Party-ID: John Smith <tel:555-2222>
Anonymity: full
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
Rseq: 9021
Content-Disposition: precondition
Contact: sip:Host(mta-t.provider)
Content-Type: application/sdp
Content-length: (.)

v=0
o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
```



```
c= IN IP4 Host(mta-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=rtpmap:0 PCMU/8000
m=audio 6544 RTP/AVP 0
a=qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, Proxy-t updates the anonymizer with the MTA-t address information:

(9) ANON_Modify:

```
ANON_Modify
Endpoint1:      Host(mta-t.provider):Port(mta-t.provider)
```

```
     Endpoint2:      Host(ann-o.provider):Port(ann-o.provider)
```

The anonymizer responds back:

```
(10) ANON_Ack:
     ANON_Ack
```

Following the anonymizer update, Proxy-t then forwards the following 183-Session-Progress message to Proxy-o, restoring the Via headers, and adding Dcs-Gate information. At this point Proxy-t has completed its transaction and does not maintain any more state for this call, processing all further signaling messages as a stateless proxy.

```
(11) 183-Session-Progress:
     SIP/2.0 183 Session Progress
     Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
     Via: SIP/2.0/UDP Host(mta-o.provider)
     Require: 100rel
     State: Host(dp-t.provider); nexthop=sip:555-2222@Host(mta-
             t.provider); gate=Host(cmts-t.provider):4321/31S14621
     State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
             o.provider); gate=Host(cmts-o.provider):3612/17S30124;
             orig-dest=tel:+1-212-555-2222; num-redirects=0
     Dcs-Gate: Host(cmts-t.provider):4321/31S14621/37FA1948
     Remote-Party-ID: John Smith <tel:+1-212-555-2222>
     Anonymity: full
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
     Rseq: 9021
     Content-Disposition: precondition
     Contact: sip:Host(ann-t.provider):Port(ann-t.provider)
     Content-Type: application/sdp
     Content-length: (.)

     v=0
```



```
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(ann-t.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
```

```
     a=rtpmap:0 PCMU/8000
     m=audio Port(ann-t.provider) RTP/AVP 0
     a=qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, Proxy-o informs the originating anonymizer about the second endpoint:

```
(12) ANON_Modify:
     ANON_Modify
     Endpoint1:      Host(mta-o.provider):Port(mta-o.provider)
     Endpoint2:      Host(ann-t.provider) :Port(ann-t.provider)
```

The anonymizer responds back:

```
(13) ANON_Ack:
     ANON_Ack
```

Subsequently, Proxy-o forwards the following 183-Session-Progress to MTA-o. At this point Proxy-o has completed its transaction and does not maintain any more state for this call, processing all further signaling messages as a stateless proxy.

```
(14) 183-Session-Progress:
     SIP/2.0 183 Session Progress
     Via: Sip/2.0/UDP Host(mta-o.provider)
     Require: 100rel
     Media-Authorization: 17S30124
     State: Host(dp-o.provider); state="{gate= Host(cmts-
             o.provider): 3612/17S30124, nexthop=sip:+1-212-555-
             2222;rn=+1-212-234-2222@Host(DP-t), state="Host(dp-
             t.provider); nexthop=sip:555-2222@Host(mta-t.provider);
             gate=Host(cmts-t.provider):4321/31S14621; orig-
             dest=tel:+1-212-555-1111; num-redirects=0"}K"
     Remote-Party-ID: <sip:{type=remote-id; orig=tel:+1-212-555-
             2222; anonymity=full}K@Host(dp-t.provider);
             user=private>;  rpi-id=private
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
     Rseq: 9021
     Content-Disposition: precondition
     Contact: sip:Host(ann-o.provider):Port(ann-o.provider)
     Content-Type: application/sdp
     Content-length: (.)
```

```
     v=0
     o=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(ann-o.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     m=audio Port(ann-o) RTP/AVP 0
     a-X=pc-qos:mandatory sendrecv confirm
```

Upon receiving the 183-Session-Progress message, MTA-o sends the following PRACK message indirectly to MTA-t through the anonymizer using the IP address in the Contact header of the 183-Session-Progress message.

```
(15) PRACK:
     PRACK sip:Host(ann-o.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 128 PRACK
     Rack: 9021 127 INVITE
     Content-Type: application/sdp
     Content-length: (.)

     v=0
     O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
     s=-
     c= IN IP4 Host(mta-o.provider)
     b=AS:64
     t=907165275 0
     a=X-pc-csuites:312F
     a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
     a=rtpmap:0 PCMU/8000
     m=audio 3456 RTP/AVP 0
     a-qos:mandatory sendrecv
```

ANN-o modifies the address information in the message and forwards it to ANN-t.

```
(16) PRACK:
     PRACK sip:Host(ann-t.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(ann-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
```

```
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 128 PRACK
Rack: 9021 127 INVITE
```



```
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(ann-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio Port(ann-o.provider) RTP/AVP 0
a-qos:mandatory sendrecv
```

ANN-t modifies the address information in the message and forwards it to MTA-t.

(17) PRACK

```
PRACK sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(ann-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 128 PRACK
Rack: 9021 127 INVITE
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(ann-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio Port(ann-t.provider) RTP/AVP 0
a-qos:mandatory sendrecv
```

MTA-t acknowledges the PRACK with a 200-OK, and begins to reserve the resources necessary for the call.

```
(18) 200 OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(ann-t.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
     seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 128 PRACK
```


ANN-t modifies the address information in the message and forwards it to ANN-o.

```
(19) 200 OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(ann-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 128 PRACK
```

ANN-o modifies the address information in the message and forwards it to MTA-o.

```
(20) 200 OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 128 PRACK
```

After sending PRACK(7), MTA-o attempts to reserve network resources if necessary. If resource reservation is successful, MTA-o sends the following COMET message directly to MTA-t. MTA-o starts timer (T-direct-request).

```
(21) COMET:
     COMET sip:Host(ann-o.provider) SIP/2.0
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
```

```
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 COMET
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(mta-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio 3456 RTP/AVP 0
a=qos:success send
```


ANN-o modifies the address information in the message and forwards it to ANN-t.

(22) COMET:

```
COMET sip:Host(ann-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(ann-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 COMET
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(ann-o.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio Port(ann-o.provider) RTP/AVP 0
a=qos:success send
```

ANN-t modifies the address information in the message and forwards it to MTA-t.

(23) COMET:

```
COMET sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(ann-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 COMET
Content-Type: application/sdp
Content-length: (.)

v=0
O=- 2987933615 2987933615 IN IP4 A3C47F2146789F0
s=-
c= IN IP4 Host(ann-t.provider)
b=AS:64
t=907165275 0
a=X-pc-csuites:312F
a=X-pc-secret:clear:WhenInTheCourseOfHumanEvents
a=rtpmap:0 PCMU/8000
m=audio Port(ann-t.provider) RTP/AVP 0
a=qos:success send
```

MTA-t acknowledges the COMET message with a 200-OK.

(24) 200 OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(ann-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 129 COMET
```

ANN-t modifies the address information in the message and forwards it to ANN-o.

(25) 200 OK:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(ann-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
```

```
     Cseq: 129 COMET
```

ANN-o modifies the address information in the message and forwards it to MTA-o.

```
(26) 200 OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 129 COMET
```

Upon receipt of the 200-OK(26), MTA-o stops timer (T-direct-request).

Upon receipt of the (17) PRACK message, MTA-t stops timer (T-proxy-response) and attempts to reserve network resources if necessary. Once MTA-t both receives the COMET message and has successfully reserved network resources, MTA-t begins to send ringing voltage to the designated line and sends the following 180 RINGING message through Proxy-t. MTA-t restarts the session timer (T3) with value (T-ringing).

```
(27) 180 RINGING:
     SIP/2.0 180 Ringing
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"; via=Host(mta-o.provider)}K
     Require: 100rel
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider); gate=Host(cmts-t.provider):4321/31S14621;
             state="Host(dp-o.provider); nexthop=sip:555-
```



```
             1111@Host(mta-o.provider); gate=Host(cmts-
             o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
             2222; num-redirects=0"}K"
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Contact: sip:Host(mta-t.provider)
     Cseq: 127 INVITE
     Rseq: 9022
```

Proxy-t decodes the Via: headers, and passes the 180-Ringing to Proxy-o. This operation is done as a SIP stateless proxy.

(28) 180 RINGING:

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
Via: SIP/2.0/UDP Host(mta-o.provider)
Require: 100rel
State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
        o.provider); gate=Host(cmts-o.provider):3612/17S30124
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Contact: sip:Host(ann-t.provider):Port(ann-t.provider)
Cseq: 127 INVITE
RSeq: 9022
```

Proxy-o handles the message as a SIP stateless proxy, and passes the 180-Ringing to MTA-o.

(29) 180 RINGING:

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP Host(mta-o.provider)
Require: 100rel
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Contact: sip:Host(ann-o.provider):Port(ann-o.provider)
Cseq: 127 INVITE
RSeq: 9022
```

Upon receipt of the 180 RINGING message, MTA-o restarts the transaction timer (T3) with value (T-ringing). MTA-o acknowledges the provisional response with a PRACK, and plays audible ringback tone to the customer.

(30) PRACK:

```
PRACK sip:Host(ann-o.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
```



```
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 130 PRACK
RAck: 9022 127 INVITE
```

ANN-o modifies the address information in the message and forwards it to ANN-t.

(31) PRACK:
PRACK sip:Host(ann-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(ann-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B; seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 130 PRACK
RAck: 9022 127 INVITE

ANN-t modifies the address information in the message and forwards it to MTA-t.

(32) PRACK:
PRACK sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(ann-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B; seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 130 PRACK
RAck: 9022 127 INVITE

MTA-t acknowledges the PRACK with a 200-OK, and stops timer (T-proxy-response).

(33) 200 OK:
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(ann-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B; seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 130 PRACK

ANN-t modifies the address information in the message and forwards it to ANN-o.

(34) 200 OK:
SIP/2.0 200 OK
Via: SIP/2.0/UDP Host(ann-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B; seq=72))@localhost>

```
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 130 PRACK
```

ANN-o modifies the address information in the message and forwards it to MTA-o.

```
(35) 200 OK:
     SIP/2.0 200 OK
     Via: SIP/2.0/UDP Host(mta-o.provider)
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 130 PRACK
```

Once MTA-t detects off-hook on the called line, it disconnects ringing voltage from the line and sends the final response through the proxies. MTA-t stops timer (T-ringing) and starts timer (T-proxy-response). If necessary, MTA-t may also commit to resources that have been reserved for this call. At this point, MTA-t begins to generate bearer channel packets of encoded voice and send them to MTA-o using the IP address and port number specified in the SDP part of the original INVITE message.

```
(36) 200-OK:
     SIP/2.0  200 OK
     Via: SIP/2.0/UDP Host(dp-t.provider), {via="Host(dp-
             o.provider); branch=1"; via=Host(mta-o.provider)}K
     State: Host(dp-t.provider); state="{nexthop=sip:Host(dp-
             o.provider); gate=Host(cmts-t.provider):4321/31S14621;
             state="Host(dp-o.provider); nexthop=sip:555-
             1111@Host(mta-o.provider); gate=Host(cmts-
             o.provider):3612/17S30124; orig-dest=tel:+1-212-555-
             2222; num-redirects=0"}K"
     From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
             seq=72))@localhost>
     To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
     Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
     Cseq: 127 INVITE
```

Proxy-t handles the message as a SIP stateless proxy, and forwards it to Proxy-o.

```
(37) 200-OK:
     SIP/2.0  200 OK
     Via: SIP/2.0/UDP Host(dp-o.provider);branch=1
     Via: SIP/2.0/UDP Host(mta-o.provider)
     State: Host(dp-o.provider); nexthop=sip:555-1111@Host(mta-
             o.provider); gate=Host(cmts-o.provider):3612/17S30124
```

```
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
```



```
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
```

Proxy-o handles the message as a SIP stateless proxy, and forwards it to MTA-o.

(38) 200-OK:

```
SIP/2.0  200 OK
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 INVITE
```

Upon receipt of the 200-OK message, MTA-o stops timer (T-ringing) and stops playing audible ringback tone to the customer and begins to play the bearer channel stream that is received from MTA-t. MTA-o sends the following ACK message to MTA-t. If necessary, MTA-o may also commit to resources that have been reserved for this call. At this point, MTA-o begins to generate bearer channel packets of encoded voice and send them to MTA-t using the IP address and port number specified in the SDP part of the original 183-Session-Progress message (that was a response to the original INVITE).

(39) ACK:

```
ACK sip:Host(ann-o.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 ACK
```

ANN-o modifies the address information in the message and forwards it to ANN-t.

(40) ACK:

```
ACK sip:Host(ann-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(ann-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
```

Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 ACK

ANN-t modifies the address information in the message and forwards it to MTA-t.

(41) ACK:
ACK sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(ann-t.provider)

From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 127 ACK

Upon receipt of the ACK message, MTA-t stop timer (T-proxy-response).

When either MTA detects hangup, it sends out a BYE message to the other MTA. In this example, MTA-o detected that the customer hung up the phone. MTA-o puts that line in the "idle" state so new calls can be made or received. It sends the following BYE message directly to MTA-t. MTA-o may also need to release network resources that have been used for the call. MTA-o starts timer (T-direct-request).

(42) BYE:
BYE sip:Host(ann-o.provider) SIP/2.0
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE

ANN-o modifies the address information in the message and forwards it to ANN-t.

(43) BYE:
BYE sip:Host(ann-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(ann-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost

```
Cseq: 131 BYE
```

ANN-t modifies the address information in the message and forwards it to MTA-t.

(44) BYE:

```
BYE sip:Host(mta-t.provider) SIP/2.0
Via: SIP/2.0/UDP Host(ann-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

Upon receipt of the BYE message, MTA-t stops playing the bearer channel stream received from MTA-o and, if necessary, releases network resources that have been used for this call. MTA-tsends the

following 200-OK message to MTA-o. MTA-t starts a 15-second timer (T-hangup) (Note: this is a local interface issue, and not part of this specification). If MTA-t does not detect hangup on the line before timer (T-hangup) expires, it plays "reorder" tone on the customer line. Once hangup is detected, MTA-t puts that line in the "idle" state so new calls can be made or received.

(45) 200-OK:

```
SIP/2.0  200 OK
Via: SIP/2.0/UDP Host(ann-t.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

ANN-t modifies the address information in the message and forwards it to ANN-o.

(46) 200-OK:

```
SIP/2.0  200 OK
Via: SIP/2.0/UDP Host(ann-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

ANN-o modifies the address information in the message and forwards it to MTA-o.

(47) 200-OK:

```
SIP/2.0  200 OK
Via: SIP/2.0/UDP Host(mta-o.provider)
From: "Alien Blaster" <sip:B64(SHA-1(555-1111; time=36123E5B;
        seq=72))@localhost>
To: sip:B64(SHA-1(555-2222; time=36123E5B; seq=73))@localhost
Call-ID: B64(SHA-1(555-1111;time=36123E5B;seq=72))@localhost
Cseq: 131 BYE
```

Upon receipt of 200-OK, MTA-o stops timer (T-direct-request).

11. Notice Regarding Intellectual Property Rights

The IETF has been notified of intellectual property rights claimed in regard to some or all of the specification contained in this document. For more information consult the online list of claimed rights.

12. References

1. Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

2. Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

3. "SIP Extensions for Caller Identity, Privacy and Operator Services", Internet Draft: <draft-ietf-sip-privacy-00.txt>, November 2000.

4. "SIP proxy-to-proxy extensions for supporting Distributed Call State", Internet Draft: <draft-dcsgroup-sip-proxy-proxy-03.txt>, November 2000.

5. "SIP extensions for supporting Distributed Call State", Internet Draft: <draft-ietf-sip-state-00.txt>, November 2000.

6. "SIP Extensions for Media Authorization", Internet Draft: <draft-ietf-sip-call-auth-00.txt>, November 2000.

7. "Reliability of Provisional Responses in SIP", Internet Draft: <draft-ietf-sip-100rel-02.txt>, July, 2000.

8. "SIP 183 Session Progress Message", Internet Draft: <draft-ietf-sip-183-00.txt>, October 1999.

9. "Integration of Resource Management and SIP for IP Telephony", Internet Draft: <draft-ietf-sip-manyfolks-resource-00.txt>, November 2000.

13. Acknowledgements

The Distributed Call Signaling work in the PacketCable project is the work of a large number of people, representing many different companies. The authors would like to recognize and thank the following for their assistance: John Wheeler, Motorola; David Boardman, Daniel Paul, Arris Interactive; Bill Blum, Jon Fellows, Jay Strater, Jeff Ollis, Clive Holborow, Motorola; Doug Newlin, Guido Schuster, Ikhlaq Sidhu, 3Com; Jiri Matousek, Bay Networks; Farzi Khazai, Nortel; John Chapman, Bill Guckel, Michael Ramalho, Cisco; Chuck Kalmanek, Doug Nortz, John Lawser, James Cheng, Tung-Hai Hsiao, Partho Mishra, AT&T; Telcordia Technologies; and Lucent Cable Communications.

14. Author's Addresses

Bill Marshall
AT&T
Florham Park, NJ 07932

Email: wtm@research.att.com

K. K. Ramakrishnan
TeraOptic Networks
Sunnyvale, CA
Email: kk@teraoptic.com

Ed Miller
Terayon
Louisville, CO 80027
Email: edward.miller@terayon.com

Glenn Russell
CableLabs
Louisville, CO 80027

Email: G.Russell@Cablelabs.com

Matt Osman
CableLabs
Louisville, CO 80027
Email: M.Osman@Cablelabs.com

Burcak Beser
Juniper Networks
Sunnyvale, CA
Email: burcak@juniper.net

Mike Mannette
3Com
Rolling Meadows, IL 60008
Email: Michael_Mannette@3com.com

Kurt Steinbrenner
3Com
Rolling Meadows, IL 60008
Email: Kurt_Steinbrenner@3com.com

Dave Oran
Cisco
Acton, MA 01720
Email: oran@cisco.com

Flemming Andreasen
Cisco
Edison, NJ
Email: fandreas@cisco.com

John Pickens
Com21
San Jose, CA
Email: jpickens@com21.com

Poornima Lalwaney

Nokia
San Diego, CA 92121
Email: poornima.lalwaney@nokia.com

Jon Fellows
Copper Mountain Networks
San Diego, CA 92121
Email: jfellows@coppermountain.com

Doc Evans
D. R. Evans Consulting
Boulder, CO 80303
Email: n7dr@arrl.net

Keith Kelly
NetSpeak
Boca Raton, FL 33587
Email: keith@netspeak.com

Full Copyright Statement

This memo is filed as <draft-dcsgroup-sip-arch-06.txt>, and expires August 31, 2002.