# Telephony:1 Security Best Practice

**For UPnP Version 1.0**
**Status: Standardized DCP (SDCP)**
**Date: March 22, 2011**
**Document Version: 1.0**
**Service Template Version: 2.00**

This Standardized DCP has been adopted as a Standardized DCP by the Steering Committee of the UPnP Forum, pursuant to Section 2.1(c)(ii) of the UPnP Forum Membership Agreement. UPnP Forum Members have rights and licenses defined by Section 3 of the UPnP Forum Membership Agreement to use and reproduce the Standardized DCP in UPnP Compliant Devices. All such use is subject to all of the provisions of the UPnP Forum Membership Agreement.

THE UPNP FORUM TAKES NO POSITION AS TO WHETHER ANY INTELLECTUAL PROPERTY RIGHTS EXIST IN THE STANDARDIZED DCPS. THE STANDARDIZED DCPS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS". THE UPNP FORUM MAKES NO WARRANTIES, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE STANDARDIZED DCPS, INCLUDING BUT NOT LIMITED TO ALL IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, OF REASONABLE CARE OR WORKMANLIKE EFFORT, OR RESULTS OR OF LACK OF NEGLIGENCE.

Copyright © 2011 UPnP Forum. All rights reserved.

Copyright © 2011 UPnP Forum. All rights reserved.

| Authors[1] | Company |
|---|---|
| Yu Zhu (Editor) | Huawei |
| Yoshiki Nishikawa | NTT |
| Jeyoung Maeng | Samsung |
| Mayuresh Patil | Samsung |
| Mahfuzur Rahman | Samsung |
| Jooyeol Lee (Chair) | Samsung |
| Enrico Grosso | Telecom Italia |
| Massimo Messore | Telecom Italia |
| Davide Moreo | Telecom Italia |
| Alessandro De Vincentis | Telecom Italia |

[1] Note: The UPnP Forum in no way guarantees the accuracy or completeness of this author list and in no way implies any rights for or support from those members listed. This list is not the specifications' contributor list that is kept on the UPnP Forum's website.

Copyright © 2011 UPnP Forum. All rights reserved.

# Contents



Copyright © 2011 UPnP Forum. All rights reserved.

# List of Tables



Copyright © 2011 UPnP Forum. All rights reserved.

# List of Figures



Copyright © 2011 UPnP Forum. All rights reserved.

# 1 Overview and Scope

This document is intended to serve as guidelines for implementing the UPnP *DeviceProtection* service [DP] – *DeviceProtection:1,* UPnP Forum, February 2011.
Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1,* UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

for UPnP Telephony version 1 Devices (i.e. *TelephonyServer:1* and *TelephonyClient:1*). Specifically, this document: recommends security policies for telephony services (e.g. default roles, and restricted set of Telephony actions, etc.), provides solutions for monopolization of calls using *DeviceProtection*, and discusses user privacy protection.

## 1.1 Introduction

UPnP Telephony enables telephony services (e.g. phone calls and messages) available to the home network. However, when telephony devices are exposed to the home network that is not trusted, there would be security threats. For example, a control point could discover a Telephony Server (TS) and make costly phone calls using the TS. Considering user privacy, a user's call logs and messages which are considered private can be read by a control point. Moreover, a malicious control point could discover conversations when a user is calling on a Telephony Client (TC) by eavesdropping of the traffic between the TS and the TC. The malicious control point could also steal user's information (e.g. accounts for online services) when the user is using input services by eavesdropping of the traffic between the input sender and receiver devices. When the home network is not trusted, the *DeviceProtection* service [DP] – *DeviceProtection:1,* UPnP Forum, February 2011.
Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1,* UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

can be implemented to the telephony devices as an option to provide secure means of using the telephony services in the home network. However, in a trusted home network, the security is not an issue since every home user trusts each other. In this document, guidelines for implementing *DeviceProtection* to UPnP Telephony devices are discussed, including:

- The default roles to be implemented, the recommended restricted telephony actions, and some other recommended security policies with regard to the implementation of the *DeviceProtection* service for the telephony specifications.
- The monopolization of calls using *DeviceProtection*.
- The protection of input data between the input sender and the receiver devices, and the voice/video communications between the TS and the TC using *DeviceProtection*.

However, security issues such as how to prevent a TS/TC or control point controlled by a malware are considered out of the scope of this document.

## 1.2 Notation

- In this document, features are described as Required, Recommended, or Optional as follows:

Copyright © 2011 UPnP Forum. All rights reserved.

The key words “MUST,” “MUST NOT,” “REQUIRED,” “SHALL,” “SHALL NOT,” “SHOULD,” “SHOULD NOT,” “RECOMMENDED,” “MAY,” and “OPTIONAL” in this specification are to be interpreted as described in [RFC 2119].

In addition, the following keywords are used in this specification:

PROHIBITED – The definition or behavior is an absolute prohibition of this specification. Opposite of REQUIRED.

CONDITIONALLY REQUIRED – The definition or behavior depends on a condition. If the specified condition is met, then the definition or behavior is REQUIRED, otherwise it is PROHIBITED.

CONDITIONALLY OPTIONAL – The definition or behavior depends on a condition. If the specified condition is met, then the definition or behavior is OPTIONAL, otherwise it is PROHIBITED.

These keywords are thus capitalized when used to unambiguously specify requirements over protocol and application features and behavior that affect the interoperability and security of implementations. When these words are not capitalized, they are meant in their natural-language sense.

- Strings that are to be taken literally are enclosed in “double quotes”.
- Words that are emphasized are printed in *italic*.
- Keywords that are defined by the UPnP Working Committee are printed using the *forum* character style.
- Keywords that are defined by the UPnP Device Architecture are printed using the **arch** character style.
- A double colon delimiter, “::”, signifies a hierarchical parent-child (parent::child) relationship between the two objects separated by the double colon. This delimiter is used in multiple contexts, for example: Service::Action(), Action()::Argument, parentProperty::childProperty.

## 1.3 References

### 1.3.1 Normative References

This section lists the normative references used in this specification and includes the tag inside square brackets that is used for each such reference:

[DEVICE] – UPnP Device Architecture, version 1.0, UPnP Forum, June 13, 2000.
Available at: http://www.upnp.org/specs/architecture/UPnP-DeviceArchitecture-v1.0-20000613.htm.
Latest version available at: http://www.upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0.pdf.

[ISO 8601] – Data elements and interchange formats – Information interchange -- Representation of dates and times, International Standards Organization, December 21, 2000.
Available at: ISO 8601:2000.

[RFC 2119] – IETF RFC 2119, Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, 1997.
Available at: http://www.faqs.org/rfcs/rfc2119.html.

[RFC 3339] – IETF RFC 3339, Date and Time on the Internet: Timestamps, G. Klyne, Clearswift Corporation, C. Newman, Sun Microsystems, July 2002.
Available at: http://www.ietf.org/rfc/rfc3339.txt.

Copyright © 2011 UPnP Forum. All rights reserved.

[XML] – Extensible Markup Language (XML) 1.0 (Third Edition), François Yergeau, Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, eds., W3C Recommendation, February 4, 2004.
Available at: http://www.w3.org/TR/2004/REC-xml-20040204.

[XML SCHEMA-2] – XML Schema Part 2: Data Types, Second Edition, Paul V. Biron, Ashok Malhotra, W3C Recommendation, 28 October 2004.
Available at: http://www.w3.org/TR/2004/REC-xmlschema-2-20041028.

### 1.3.2 Informative References

This section lists the informative references that are provided as information in helping understand this specification:

[DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

[MMS] – *MediaManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/ phone/UPnP-phone-MediaManagement-v1-Service-20110322.pdf.

Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-MediaManagement-v1-Service.pdf.

[ICS] – *InputConfig:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-InputConfig-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-InputConfig-v1-Service.pdf.

[RFC 3550] – IETF RFC 3550, RTP: A Transport Protocol for Real-Time Applications, H. Schulzrinne, Columbia University, S. Casner, Packet Design, R. Frederick Blue Coat Systems Inc.,V. Jacobson, Packet Design, July 2003.
Available at: http://www.ietf.org/rfc/rfc3550.txt.

[RFC 3711] – IETF RFC 3711, The Secure Real-time Transport Protocol (SRTP), M. Baugher, D. McGrew, Cisco Systems, Inc., M. Naslund, E. Carrara, K. Norrman, Ericsson Research, March 2004.
Available at: http://www.ietf.org/rfc/rfc3711.txt.

[RFC 4566] – IETF RFC 4566, SDP: Session Description Protocol, M. Handley, UCL, V. Jacobson, Packet Design, C. Perkins, University of Glasgow, July 2006.
Available at: http://www.ietf.org/rfc/rfc4566.txt.

[RFC 4568] – IETF RFC 4568, Session Description Protocol (SDP) Security Descriptions for Media Streams, F. Andreasen, M. Baugher, D. Wing, Cisco Systems, July 2006.
Available at: http://www.ietf.org/rfc/rfc4568.txt.

## 1.4 Terms and Abbreviations

### 1.4.1 Abbreviations

**Table 1-2: Abbreviations**

| Definition | Description |
|---|---|
| CaMS | CallManagement Service |

Copyright © 2011 UPnP Forum. All rights reserved.

| Definition | Description |
|---|---|
| MMS | MediaManagement Service |
| DP | DeviceProtection Service |
| TS | Telephony Server |
| TC | Telephony Client |
| TelCP | Telephony Control Point |
| ICS | InputConfig Service |
| ICP | Input Control Point |

## 1.4.2 Terms

### 1.4.2.1 Telephony Server (TS)

The term Telephony Server (TS) refers to a logical device that provides common telephony features (e.g. call/video call, messaging, and address book) via UPnP to other devices in the home network. A TS is usually connected to a telephony service on its WAN interface, either wire line or mobile. For example, a TS may be a mobile phone or a home gateway with VoIP features.

### 1.4.2.2 Telephony Client (TC)

The term Telephony Client (TC) to a networked logical device that allows the user to enjoy the telephony features provided by the Telephony Server via UPnP. A TC may usually provide input/output features for voice and video. An example of a TC is a networked TV Set.

### 1.4.2.3 Telephony Control Point (TelCP)

The term Telephony Control Point (TelCP) refers to a software feature able to control the functionalities of both TS and TC. It may be embedded in a TS, a TC or also being a physical device on its own.

### 1.4.2.4 InputConfig Control Point (ICP)

The Term InputConfig Control Point (ICP) refers to a software feature that is able to control the functionalities of UPnP devices to be used to provide user-friendly input features. The control here refers to getting capabilities of UPnP devices to be used for input, matching capabilities and selecting the appropriate device role such as receiving side or sending side, etc.

### 1.4.2.5 InputConfig Service (ICS)

The Term InputConfig Service (ICS) refers to a software feature that is able to provide user-friendly input capability via UPnP means and expose interfaces to describe capabilities of sender/receiver of devices to be used for input services and setup the input session between the devices using the matching profile (capability) from the ICP.

### 1.4.2.6 Monopolize

Monopolize indicates the process for an identity (a TelCP or a user) to obtain the exclusive rights to control a specific call managed by the *CallManagement* service. When a call is successfully Monopolized by an identity, the *CallManagement* service only allows this specific identity to manage the call, until the call is terminated or this identity hands over the exclusive rights to control the call to another identity.

Copyright © 2011 UPnP Forum. All rights reserved.

#### 1.4.2.7 Monopolizer

Monopolizer indicates the identity (a TelCP or a user) that currently has the exclusive rights to control a specific call managed by the *CallManagement* service.

#### 1.4.2.8 Call Monopolization

Call Monopolization indicates the mechanisms that allow an identity (a TelCP or a user) to Monopolize a Call managed by the *CallManagement* service. Refer to the *CallManagement* service [CaMS] section 2.3.2 for detail.

Copyright © 2011 UPnP Forum. All rights reserved.

# 2 Security Policies

## 2.1 Access Control and User Roles

The *TelephonyServer:1* and *TelephonyClient:1* that implement the *DeviceProtection* service implement the following user roles as defined in [DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

:

- *Public* REQUIRES neither authentication nor authorization. This user role is intended for telephony actions that do no harm to telephony devices and retrieves no user private information, e.g. discovery of media/messaging capabilities of TS or TC.
- *Basic* REQUIRES authentication and authorization. Authentication is based on *DeviceProtection*
  [DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
  Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
  Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

- . This user role is intended for general telephony operations such as making a phone call, sending/reading messages, and sending input data etc. These operations may expose user privacy and cause service charges.
- *Admin* REQUIRES authentication and authorization. Authentication is based on *DeviceProtection*
  [DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
  Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
  Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

- . This user role is intended for managing user/control point identities and roles in the context of *DeviceProtection*. In the implementation of telephony services, any call can be managed by the *Admin* role.
- *Vendor-defined* roles can be created, but it is REQUIRED that all other roles are implemented and supported. It is also REQUIRED that control points supporting three predefined roles are able to operate with the Telephony Server and Telephony Client devices. Role names MUST be maximum 64 characters long, without any spaces and MUST NOT contain spaces. Role names not defined by the Forum MUST be prefixed, vendor-specific Names with a Vendor Domain Name followed by a colon (such as “example.com:”). Forum-defined Role names MUST be defined in service specifications and/or DCP-specific security considerations documents published by Working Committees.

Copyright © 2011 UPnP Forum. All rights reserved.

This document gives recommendations as to which set of Telephony actions can be restricted to unknown control points. In addition, it is up to the implementation to decide what access level is required to invoke a specific restricted telephony action.

In this document, it is also up to the implementation to select which default role to be automatically granted to a control point after pairwise (i.e. WPS) introduction to a Telephony Server or a Telephony Client device. The *Admin* level role is granted as defined in the *DeviceProtection* [DP] – *DeviceProtection:1,* UPnP Forum, February 2011.
Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1,* UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

. Examples for granting default roles to an introduced control point are listed as follows:

- *Basic* role granted after introduction. A TelCP (e.g. a user's laptop) can be granted this role after introduction by a telephony device (e.g. VoIP home gateway) that is shared within the home network. Since the telephony services provided by this shared device are available for any home member, the *Basic* role can be assigned to the introduced TelCPs permanently for easy of usage.
- *Public* role granted after introduction, with the control point's ID ("Control Point Identity" as defined in *DeviceProtection*) presented in device's ACL. A TelCP (e.g. living room TV) that is shared within the home network can be granted this role by a user private device (e.g. a user's personal smartphone) after introduction. A home member can access his/her own device after user login using this shared TelCP.
- *Vendor-defined* role granted after introduction. Vendors can define additional role and associate it with a control point that has introduced itself to the TS or the TC.

## 2.2 General Policies

*DeviceProtection* is an OPTIONAL service which can be implemented in *TelephonyServer:1* and/or *TelephonyClient:1* devices.

Depending on different usages, The *Admin* role MAY be allowed to manage any outgoing/incoming call. Vendors MAY also allow user identities and roles only, i.e. every time user login is required to perform restricted telephony operations (e.g. making calls, sending/reading messages, etc.).

## 2.3 Security Policies for Telephony Services

The sections below list all actions of UPnP Telephony version 1 Services, and provide recommendations on whether an action is restricted or not. In the tables below, the meanings of the terms "*non-restrictable*" and "*restrictable*" are:

- *non-restrictable*: indicates an action is RECOMMENDED to be invoked by any control point including unknown ones, i.e. the action is allowed to be invoked without TLS tunnel.
- *restrictable*: indicates an action is RECOMMENDED to be invoked by only authorized control points (e.g. whose role is *Basic* or *Admin*, or *Public* but has been introduced), i.e. the action is RECOMMENDED to be invoked in TLS tunnel.

Error Code 606 "Action not authorized" is returned when a control point that does not have the required role invokes a *restrictable* action of any Telephony version 1 services. However, implementations MAY also allow unauthorized control points (e.g., with *Public* role) to invoke a *restrictable* action successfully, but the returned argument values would be different from or subset of the same action invoked by an authorized control point via TLS tunnel.

Copyright © 2011 UPnP Forum. All rights reserved.

Note that the following tables list whether an action is RECOMMENDED to be restricted or not, vendors SHOULD associate proper role(s) with each action when implementing Telephony version 1 Services with *DeviceProtection*. Moreover, implementations are RECOMMENDED to follow the security policies for the *DeviceProtection* actions as specified in the *DeviceProtection* [DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

specification, thus the actions of the *DeviceProtection* service are not listed in the following tables.

### 2.3.1 *CallManagement:1*

The following table lists the recommendations on whether an action in the *CallManagement:1* service is restrictable or not.

**Table 2-1: *CallManagement:1* Actions**

| Name | Category |
|---|---|
| *GetTelephonyIdentity()* | *Restrictable* |
| *RegisterTelCPName()* | *Restrictable* |
| *UnregisterTelCPName()* | *Restrictable* |
| *ChangeTelCPName()* | *Restrictable* |
| *GetTelCPNameList()* | *Non-restrictable* |
| *GetMediaCapabilities()* | *Non-restrictable* |
| *StartCall()* | *Restrictable* |
| *StopCall()* | *Restrictable* |
| *AcceptCall()* | *Restrictable* |
| *RejectCall()* | *Restrictable* |
| *ModifyCall()* | *Restrictable* |
| *AcceptModifyCall()* | *Restrictable* |
| *StartMediaTransfer()* | *Restrictable* |
| *ChangeMonopolizer()* | *Restrictable* |
| *InitiateCall()* | *Restrictable* |
| *GetCallInfo()* | *Restrictable* |
| *GetCallLogs()* | *Restrictable* |
| *ClearCallLogs()* | *Restrictable* |
| *RegisterCallBack()* | *Restrictable* |
| *ClearCallBack()* | *Restrictable* |
| *GetCallBackInfo()* | *Restrictable* |

Copyright © 2011 UPnP Forum. All rights reserved.

### 2.3.2 *MediaManagement:1*

The following table lists the recommendations on whether an action in the *MediaManagement:1* service is restrictable or not.

**Table 2-2: *MediaManagement:1* Actions**

| Name | Category |
|---|---|
| *GetMediaCapabilities()* | *Non-restrictable* |
| *StartMediaSession()* | *Restrictable* |
| *StopMediaSession()* | *Restrictable* |
| *ModifyMediaSession()* | *Restrictable* |
| *GetMediaSessionInfo()* | *Non-restrictable* |

### 2.3.3 *Messaging:1*

The following table lists the recommendations on whether an action in the *Messaging:1* service is restrictable or not.

**Table 2-3: *Messaging:1* Actions**

| Name | Category |
|---|---|
| *GetTelephonyIdentity()* | *Restrictable* |
| *GetMessagingCapabilities()* | *Non-restrictable* |
| *GetNewMessages()* | *Restrictable* |
| *SearchMessages()* | *Restrictable* |
| *ReadMessage()* | *Restrictable* |
| *SendMessage()* | *Restrictable* |
| *DeleteMessage()* | *Restrictable* |
| *CreateSession()* | *Restrictable* |
| *ModifySession()* | *Restrictable* |
| *AcceptSession()* | *Restrictable* |
| *GetSessionUpdates()* | *Restrictable* |
| *GetSessions()* | *Restrictable* |
| *JoinSession()* | *Restrictable* |
| *LeaveSession()* | *Restrictable* |
| *CloseSession()* | *Restrictable* |
| *StartFileTransfer()* | *Restrictable* |
| *CancelFileTransfer()* | *Restrictable* |
| *GetFileTransferSession()* | *Restrictable* |

Copyright © 2011 UPnP Forum. All rights reserved.

### 2.3.4 *InputConfig:1*

The following table lists the recommendations on whether an action in the *InputConfig:1* service is restrictable or not.

**Table 2-4: *InputConfig:1* Actions**

| Name | Category |
|---|---|
| *GetInputCapability()* | *Non-restrictable* |
| *GetInputConnectionList()* | *Non-restrictable* |
| *SetInputSession()* | *Restrictable* |
| *StartInputSession()* | *Restrictable* |
| *StopInputSession()* | *Restrictable* |
| *SwitchInputSession()* | *Restrictable* |
| *SetMultiinputMode()* | *Restrictable* |
| *SetMonopolizedSender()* | *Restrictable* |

### 2.3.5 *ConfigurationManagement:1*

The following table lists the recommendations on whether an action in the *ConfigurationManagement:1* service is restrictable or not.

**Table 2-5: *ConfigurationManagement:1* Actions**

| Name | Category |
|---|---|
| *GetSupportedDataModels()* | *Non-restrictable* |
| *GetSupportedParameters()* | *Non-restrictable* |
| *GetInstances()* | *Restrictable* |
| *GetValues()* | *Restrictable* |
| *GetSelectedValues()* | *Restrictable* |
| *SetValues()* | *Restrictable* |
| *CreateInstance()* | *Restrictable* |
| *DeleteInstance()* | *Restrictable* |
| *GetAttributes()* | *Restrictable* |
| *SetAttributes()* | *Restrictable* |
| *GetInconsistentStatus()* | *Non-restrictable* |
| *GetConfigurationUpdate()* | *Non-restrictable* |
| *GetCurrentConfigurationVersion()* | *Non-restrictable* |
| *GetSupportedDataModelsUpdate()* | *Non-restrictable* |
| *GetSupportedParametersUpdate()* | *Non-restrictable* |
| *GetAttributeValuesUpdate()* | *Non-restrictable* |

Copyright © 2011 UPnP Forum. All rights reserved.

# 3 Call Monopolization using DeviceProtection

In the *CallManagement* [CaMS] service section 2.3.2, concept and solution for Call Monopolization are discussed. In particular, the mechanism defined there called "PHONE-based Call Monopolization" does not utilize the *DeviceProtection* service. In this section, *DeviceProtection* based mechanism for Call Monopolization (named herein "DP-based Call Monopolization") is described. This section also provides security considerations for "PHONE-based Call Monopolization" mechanism.

Call Monopolization as discussed in the *CallManagement* service can be TelCP-level or User-level. In the *CallManagement* service, Call Monopolization is TelCP-level, allowing a call to be Monopolized by a specific TelCP identified by its *TelCPName*. The DP-based Call Monopolization as discussed in this section targets both TelCP-level and User-level, i.e. allowing a specific TelCP or a user to Monopolize a call. When the TS implements the *DeviceProtection* service, Call Monopolization can be achieved by collaboration between the *DeviceProtection* service (DP) and the *CallManagement* service (CaMS) as internal interaction of the TS device. DP passes the TelCP/User identity and its Role associated to the current TLS session between TS and TelCP to CaMS. CaMS uses this information to decide if a new call is successfully Monopolized (e.g. for *StartCall()* and *AcceptCall()* actions) by the identity, or if the identity is authorized to manage an existing Monopolized call (e.g. for *ModifyCall()* and *StopCall()* actions).

## 3.1 TelCP-Level Call Monopolization

TelCP-level Call Monopolization allows a specific TelCP (identified by the "Control Point Identity" as defined in *DeviceProtection*) to Monopolize a call. When a TelCP invokes the *StarCall()* or *AcceptCall()* action via TLS tunnel to Monopolize a call, the TS can recognize the TelCP's identity and maps it to the *CallID* of this call. Following figure and text illustrate the operations for TelCP-level Call Monopolization.

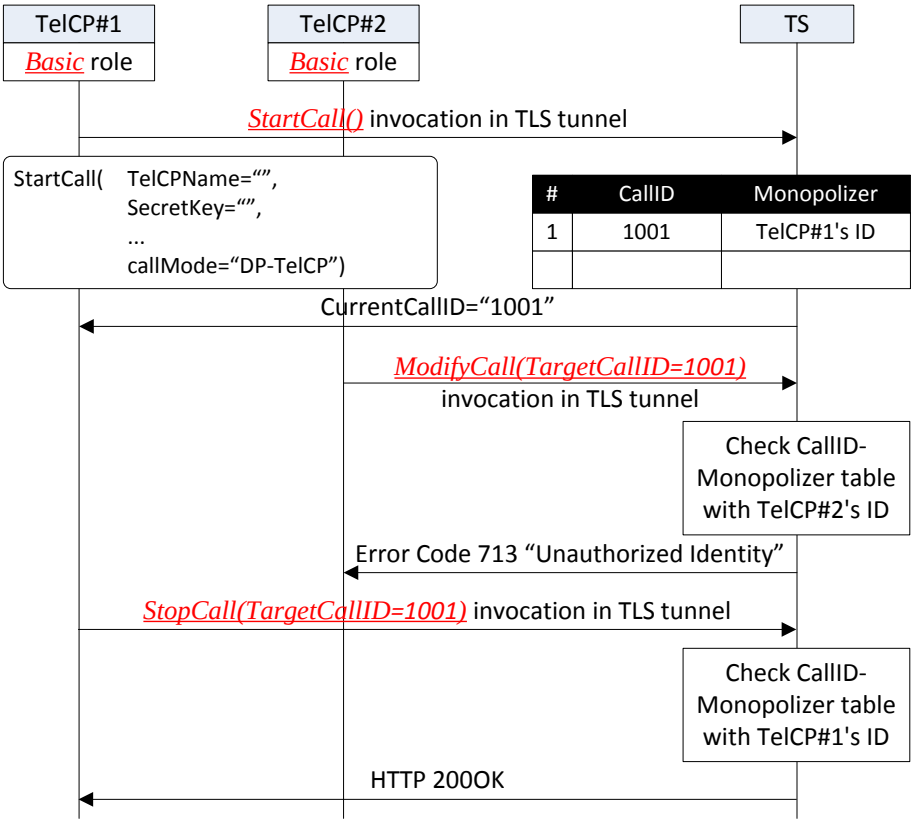



**Figure 1: DP-based TelCP-level Call Monopolization**

Copyright © 2011 UPnP Forum. All rights reserved.

1. Before invoking the *CallManagement* actions on the TS, every TelCP should be introduced to the TS via direct or indirect introduction as described in *DeviceProtection* [DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
   Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
   Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1,* UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

2. , and be granted proper role (e.g. *Basic* or *Public*).
3. TelCP#1 which has *Basic* role on TS invokes *StartCall()* action to initiate an outgoing call. In particular,
   - The *CallMode* input argument is set to "DP-TelCP", indicating TelCP#1 will Monopolize the outgoing call.
4. The TS returns the generated *CallID* to TelCP#1, and maps TelCP#1's ID ("Control Point Identity" as defined in *DeviceProtection*) to the *CallID*. The "CallID-Monopolizer" mapping table will be used for access control on the following call management actions from TelCPs. For example,
   - TelCP#2 which also has *Basic* role on the TS invokes the *ModifyCall()* action, the TS checks "CallID-Monopolizer" mapping table which indicates TelCP#2's is not the Monopolizer of the call. Thus, the *CallManagement* service returns Error Code 713 "Unauthorized Identity" to TelCP#2 indicating access denied.
   - TS implementation can decide if an *Admin* role is authorized to manage any Monopolized call.
5. Current Monopolizer (i.e. TelCP#1) of the call is allowed to perform management functions (e.g. *ModifyCall()* or *StopCall()*) of this call afterwards. For example,
   - When *StopCall()* action is invoked by TelCP#1, TS checks "CallID-Monopolizer" mapping table with TelCP#1's ID and the *CallID*, and then permits the operation.

### 3.1.1 Incoming Call Handling (TelCP-Level)

When the TS receives a particular incoming call, it can decide by its internal policies that only subsets of TelCP(s) are allowed to answer the call. Through internal interfaces, the *CallManagement* service is able to collect a list of TelCP IDs (i.e. "Control Point Identity" as defined in *DeviceProtection*) from the ACL data managed by the *DeviceProtection* service. When the *CallManagement* service receives an incoming call, it can decide a subset of TelCP(s) that are allowed to answer the call by listing their IDs in the evented *CallInfo* state variable of this call. The TelCP(s) are required to invoke the *AcceptCall()* or *RejectCall()* action via TLS tunnel to accept/reject the call, thus the TS can recognize the TelCP's identity and permit the operation. Following figure and text illustrate how to handle incoming calls.

Copyright © 2011 UPnP Forum. All rights reserved.

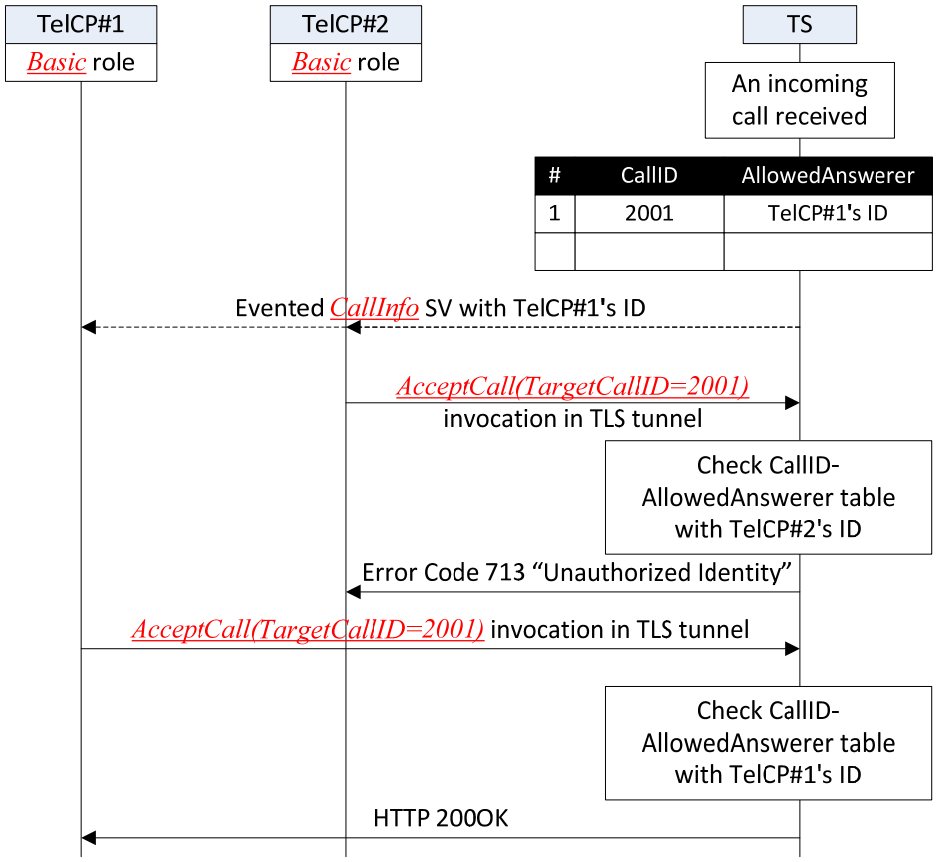



**Figure 2: Incoming call handling (TelCP-level)**

1. Before invoking the *CallManagement* actions on the TS, every TelCP should be introduced to the TS via direct or indirect introduction as described in *DeviceProtection* [DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
   Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
   Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

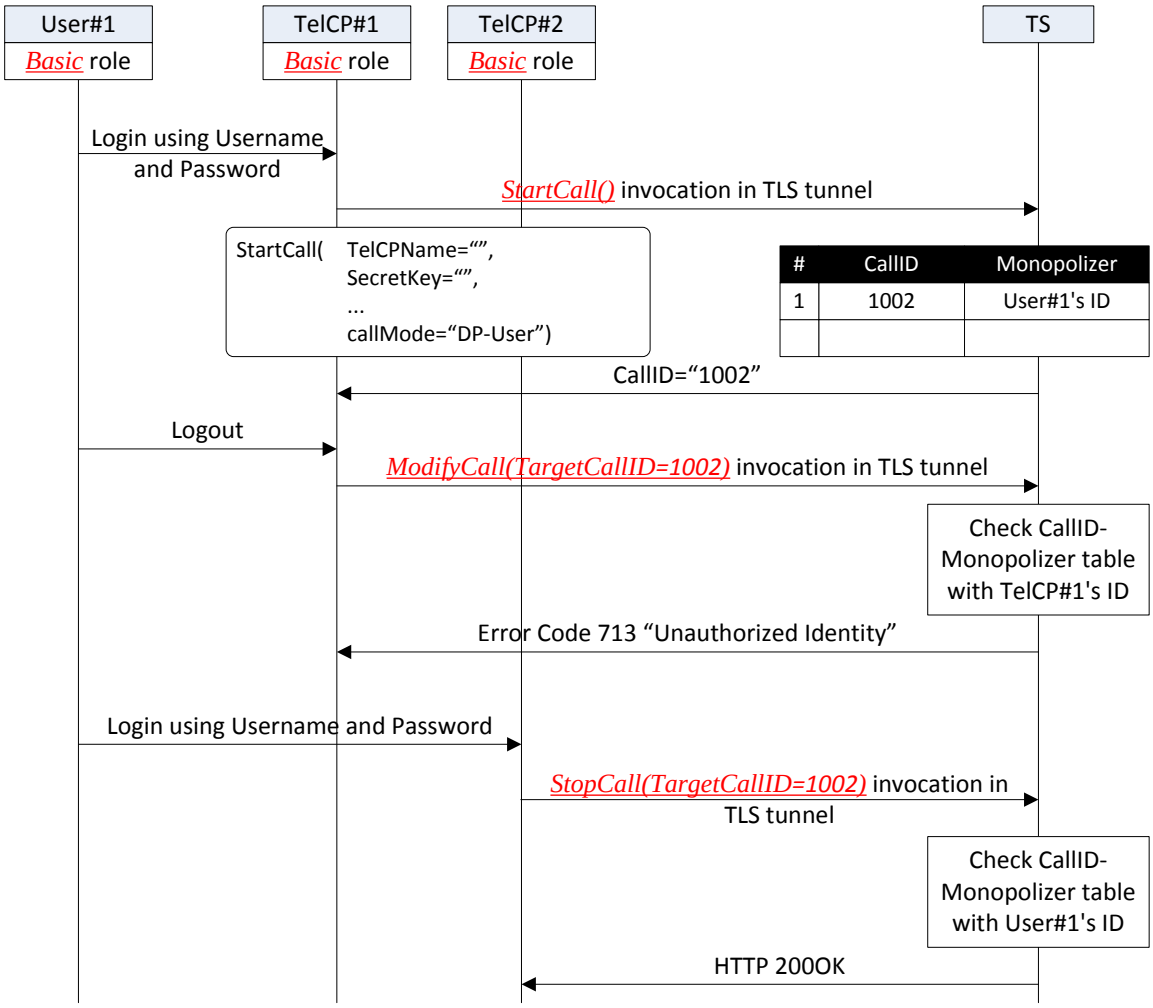
2. , and be granted proper role (e.g. *Basic* or *Public*).
3. When the TS receives an incoming call, it decides one or more TelCPs that are allowed to answer or reject the call by listing their IDs (i.e. "Control Point Identity" as defined in *DeviceProtection*) in the evented *CallInfo* state variable which is sent to all the subscribed TelCPs.
   - This indicates that before the *CallInfo* event is sent out, the *CallManagement* service is able to get out a list of TelCP identities from the ACL data managed by the *DeviceProtection* service via internal interfaces.
   - As in the figure above, TS maintains "CallID-AllowedAnswerer" mapping table internally which will be used for access control on *AcceptCall()* or *RejectCall()* actions targeting the incoming call from TelCPs.

Copyright © 2011 UPnP Forum. All rights reserved.

4. On receiving the *CallInfo* event with TelCP IDs list, the TelCPs that understand *DeviceProtection* will recognize the allowed answerer of the call. TelCPs with user interface should display the information of those TelCP(s) (e.g. name and/or alias, etc.) with ringing tone.
5. The incoming call can be answered by the allowed TelCPs by invoking the *AcceptCall()* action. For example, as shown in the figure above,
    - TelCP#2 which is not the allowed answerer will fail in invoking *AcceptCall()* action to answer the incoming call.
    - TelCP#1 invokes *AcceptCall()* action via TLS tunnel to answer the call, the TS checks the "CallID-AllowedAnswerer" table with the *CallID* and TelCP#1's ID and then permits the operation.
    - The TS implementation can decide if the *Admin* role is authorized to answer or reject any incoming call, no matter if it is the allowed answerer or not.

## 3.2 User-Level Call Monopolization

User-level Call Monopolization allows a specific User (identified by the "Username" as defined in *DeviceProtection*) to Monopolize a call using any authorized TelCP. User-level Call Monopolization requires User login before the TelCP invokes *StartCall()* or *AcceptCall()* action via TLS tunnel to Monopolize a call, then the TS can recognize the User identity and maps it to the *CallID* of the call. Following figure and text illustrate the operations for User-level Call Monopolization.




Copyright © 2011 UPnP Forum. All rights reserved.

**Figure 3: DP-based User-level Call Monopolization**

1. Before invoking the *CallManagement* actions on the TS, every TelCP should be introduced to the TS via direct or indirect introduction as described in *DeviceProtection* [DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
   Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
   Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

2. , and be granted proper role (e.g. *Basic* or *Public*).
3. User#1 which has *Basic* role on the TS logs onto TelCP#1 as defined in *DeviceProtection*, so the current TLS session between TelCP#1 and the TS is associated with User#1's identity (i.e. Username as defined in *DeviceProtection*) and role (i.e. *Basic* in the figure above).
4. TelCP#1 invokes *StartCall()* action to initiate an outgoing call. In particular,
   - The *CallMode* input argument is set to "DP-User", indicating User#1 will Monopolize the outgoing call.
5. TS returns the generated *CallID* to TelCP#1, and internally maps User#1's identity (i.e. Username) to the *CallID*. This "CallID-Monopolizer" mapping table will be used for access control on subsequent call management actions from TelCPs. For example,
   - After User#1 logging out from TelCP#1, TelCP#1 is no longer authorized to manage the call, i.e. any intent to manage the call will be rejected with Error Code 713 "Unauthorized Identity" by the *CallManagement* service.
   - TS implementation can decide if an *Admin* role is authorized to manage any Monopolized call.
6. Current Monopolizer (i.e. User#1) of the call is allowed to perform management functions (e.g. *ModifyCall()* or *StopCall()*) of this call afterwards by logging onto any authorized TelCP (e.g. TelCP#2 in the figure above). For example,
   - When *StopCall()* action is invoked by TelCP#2 with User#1 logged in, TS checks "CallID-Monopolizer" mapping table with User#1's Username and the *CallID*, and then permits the operation.

### 3.2.1 Incoming Call Handling (User-Level)

When the TS receives a particular incoming call, it can decide by its internal policies that only subsets of User(s) are allowed to answer the call. Through internal interfaces, the *CallManagement* service is able to collect a list of User IDs (i.e. "Username" as defined in *DeviceProtection*) from the ACL data managed by the *DeviceProtection* service. When the *CallManagement* service receives an incoming call, it can decide a subset of User(s) that are allowed to answer the call by listing their IDs in the evented *CallInfo* state variable of this call. The User(s) must login on to a TelCP and then invoke the *AcceptCall()* or *RejectCall()* action via TLS tunnel to accept/reject the call, thus the TS can recognize the User's identity and permit the operation. Following figure and text illustrate how to handle incoming calls.

Copyright © 2011 UPnP Forum. All rights reserved.

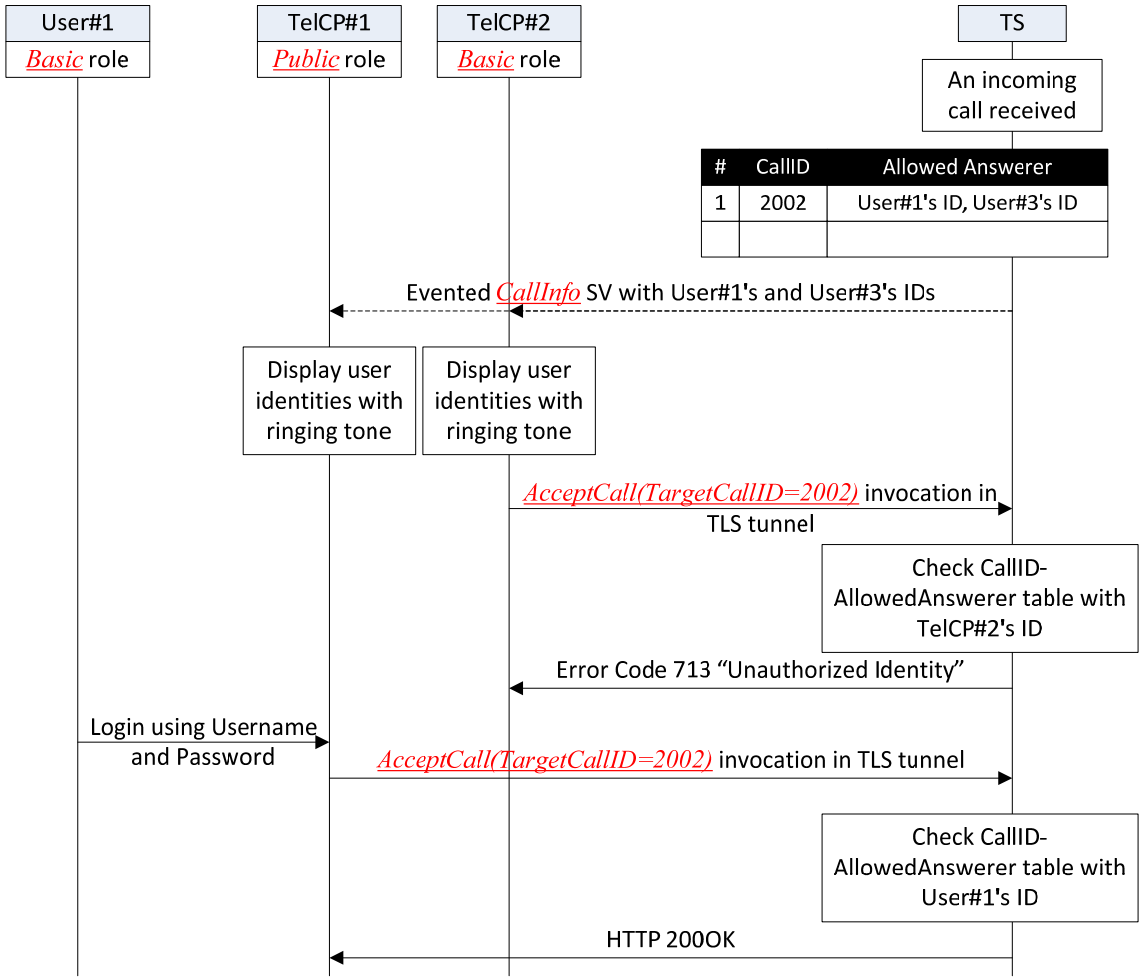



**Figure 4: Incoming call handling (User-level)**

1. Before invoking the *CallManagement* actions on the TS, every TelCP should be introduced to the TS via direct or indirect introduction as described in *DeviceProtection* [DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
   Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
   Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

2. , and be granted proper role (e.g. *Basic* or *Public*).
3. When TS receives an incoming call, it decides one or more users that are allowed to answer the call by listing their IDs (i.e. "Username" as defined in *DeviceProtection*) in the evented *CallInfo* state variable, which is sent to all the subscribed TelCPs.
   - This indicates that before the *CallInfo* event is sent out, the *CallManagement* service is able to get out a list of User identities from the ACL data managed by the *DeviceProtection* service via internal interfaces.

Copyright © 2011 UPnP Forum. All rights reserved.

- As in the figure above, TS maintains “CallID-AllowedAnswerer” mapping table internally which will be used for access control on *AcceptCall()* or *RejectCall()* actions targeting the incoming call from TelCPs.

4. On receiving the *CallInfo* event with User IDs (i.e. Username) list, the TelCPs that understand *DeviceProtection* will recognize the allowed answerer of the call. TelCPs with user interface should display the information of those user(s) (e.g. name and/or alias, etc.) with ringing tone.
5. The incoming call can be answered by the allowed user(s) by performing user login onto any authorized TelCP and then invoke the *AcceptCall()* action. For example, as shown in the figure above,
   - Without User#1 or User#3 login, neither TelCP#1 nor TelCP#2 is authorized to answer the incoming call, i.e. Error Code 713 “Unauthorized Identity” will be returned by the *CallManagement* service for *AcceptCall()* or *RejectCall()* action targeting the incoming call.
   - After User#1 logging onto TelCP#1, TelCP#1 will succeed in invoking *AcceptCall()* or *RejectCall()* action to answer or reject that incoming call.
   - The TS implementation can decide if the *Admin* role is authorized to answer or reject any incoming call, no matter if it is the allowed answerer or not.

## 3.3 Secure PHONE-based Call Monopolization

The “PHONE-based Call Monopolization” mechanism as defined in the *CallManagement* service enables the service to use the defined *TelCPName* and *SecretKey* to identify TelCPs and make Call Monopolization decisions. The mechanism can be used by the TS with or without implementing the *DeviceProtection* service. However, the mechanism depends on the confidentiality of the *SecretKey* of a TelCP, which if obtained by a malicious control point, may result in impersonation of the victim TelCP by the malicious control point. Thus, the malicious control point could e.g. modify/terminate any ongoing call Monopolized by the victim TelCP or even deregister the victim TelCP from the TS. With *DeviceProtection* implemented by the TS, a TelCP can invoke the *CallManagement* actions (e.g. *RegisterTelCPName()*, *ChangeMonopolizer()*, and *StartCall()*, etc) which contain the *SecretKey* element via TLS tunnel, so as to deliver the *SecretKey* of the TelCP securely between the TelCP and the TS.

Copyright © 2011 UPnP Forum. All rights reserved.

# 4 Secure Media Sessions

One significant security consideration is the protection of user privacy. For instance, a user may demand a private call to one of his/her contacts, or use a phone-banking service with a TC in the home. In these scenarios, a very high level of security (e.g. encryption and authentication) is required. This section focuses on establishment of a secure media session between the TS and the TC by an authorized TelCP using the *DeviceProtection* service. In particular, this document only addresses securing A/V call sessions between the TS and the TC. Note that since the communication in the WAN side is out of the scope of telephony specification, this document only addresses the security within the home network.

## 4.1 Secure A/V Call Session between TS and TC

In the case of initiating a private A/V call or using the phone-banking service using a TC, the user will expect the data transported between the TS and the TC to be secured. In the case of a UDP-based (e.g. RTP [RFC 3550]) media session between the TS and the TC, the SRTP [RFC 3711] is the best candidate for securing the data transfer between the TS and the TC. The SRTP protocol adds a security (i.e. confidentiality, authentication and integrity) to the RTP packets. The RFC 4568 [RFC 4568] provides the SDP [RFC 4566] attribute "crypto" to carry a keying information (i.e., cryptographic algorithms and master key(s) for SRTP session). The Telephony Control Point (TelCP) is accountable for setting up the media session between the TS and the TC that support the *CallManagement* service [CaMS] and the *MediaManagement* service [MMS] – *MediaManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/ phone/UPnP-phone-MediaManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-MediaManagement-v1-Service.pdf.

[ICS] – *InputConfig:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-InputConfig-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-InputConfig-v1-Service.pdf.

respectively. The figure and sequence text below illustrate the procedures for the TelCP to set up a SRTP session between the TS and the TC.

Copyright © 2011 UPnP Forum. All rights reserved.

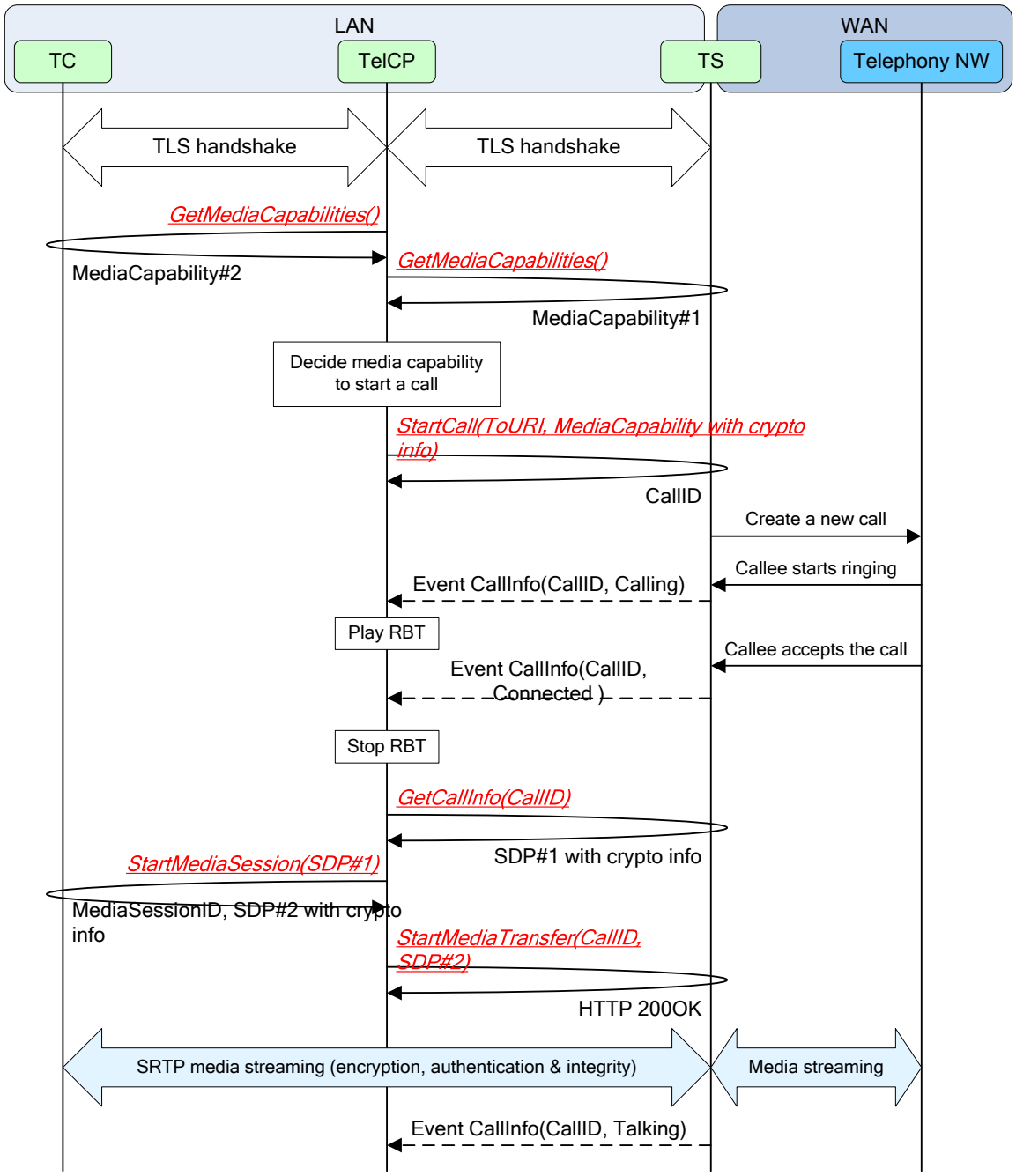



**Figure 5: SRTP-based media session set up by a TelCP**

The User will use the Telephony Control Point (TelCP) to initiate the action on the TS to make a call to the callee in the WAN telephony network, and set up an SRTP media session between the TS and the TC.

1. Before invoking any restricted action on the TS and the TC, the TelCP sets up TLS connections with the TS and the TC respectively in order to deliver the SRTP keying info securely to the TS and the TC. This assumes that the TS and the TC have assigned proper role to the TelCP before the actions are invoked. Once the TLS handshake is successfully completed, any messages containing the keying info between the TelCP and the TS or between the TelCP and the TC should be encrypted and integrity protected.

Copyright © 2011 UPnP Forum. All rights reserved.

2. The TelCP invokes all the actions via the TLS tunnels for initiating the Call. First the TelCP invokes the *GetMediaCapability()* actions on the TS and the TC to get the supported media capabilities. The TelCP will decide common media capabilities that are supported by the TS and the TC both. If both the TS and the TC support the SRTP, then the TelCP generates the crypto info (i.e. crypto suite and master key(s), and optionally SRTP session parameters) suitable for the SRTP session between the TS and the TC and delivers this information to the TS in the input argument *MediaCapabilityInfo* of the *StartCall()* action. An example of *MediaCapabilityInfo* is as follows (only elements relevant to crypto info described using SDP are shown):

```
v=0
o=- 0 0 IN IP4 192.168.0.10
s=
c=IN IP4 192.168.0.10/127
t=0 0
m=audio 0 RTP/SAVP 0
a=rtpmap:0 PCMU/8000
a=crypto:1 AES_CM_128_HMAC_SHA1_80
inline:ACHfX28zPW1acdkKCEkgeiopMaJ9fQQ1eDVACEVz|1:4
```

   In this example, the TelCP instructs the TS to start a SRTP media session with the TC, use AES counter mode for message encryption, and use the HMAC SHA1 for message authentication and integrity protection. The master key and the salt are included so that the TS can generate session keys for the encryption and the authentication for the RTP packets.

3. The TS initiates a call to the callee in the WAN side using telephony protocol e.g. SIP. After the Call is successfully connected with the Callee, the TS generates the *CallInfo* event containing *CallID* and the status of the Call to the TelCP. Then the TelCP invokes the *GetCallInfo()* action to get the detail Call information like Agreed Media Capabilities. The crypto info presented in the SDP#1 is same as provided by the TelCP in the *StartCall()* action. This crypto info will be used by the TS during the media session for encrypting and integrity protecting the RTP and RTCP messages before sending to the TC.

   - If the TS is capable and willing to generate and manage master keys, the TS can use its own master key(s) by inserting it into the SDP#1 rather than using the key(s) provided by the TelCP.

4. The TelCP forwards the TS MediaCapability (SDP#1) information to the TC by invoking the *StartMediaSession()* action on the TC. The crypto info presented in SDP#1 will be used by the TC to decrypt and integrity-check all SRTP and SRTCP packets coming from the TS during the media session. The TC returns its own MediaCapability (SDP#2) according to the media parameters given by the SDP#1 in the response of the *StartMediaSession()* action. The master key(s) in the SDP#2 can be the same as given in the SDP#1, which means the TS and the TC will use the same master key(s). The TC will use the crypto info presented in the SDP#2 to encrypt and integrity protect the RTP and the RTCP messages before sending it to the TS.

   - If the TC is capable and willing to generate and manage master keys, the TC can use its own master key(s) by inserting it into the SDP#2 rather than using the key(s) provided by the TS.

5. The TelCP forwards the TC MediaCapability (SDP#2) information to the TS by invoking the *StartMediaTransfer()* action on the TS. The crypto info presented in SDP#2 will be used by the TS to decrypt and integrity-check all SRTP and SRTCP packets coming from the TC during the media session.

6. The TC and the TS start the media session and send the SRTP packets and the SRTCP reports to each other. All the messages are encrypted and integrity protected using the keys and the cryptographic algorithms derived from the crypto info presented in the SDP#1 and the SDP#2.

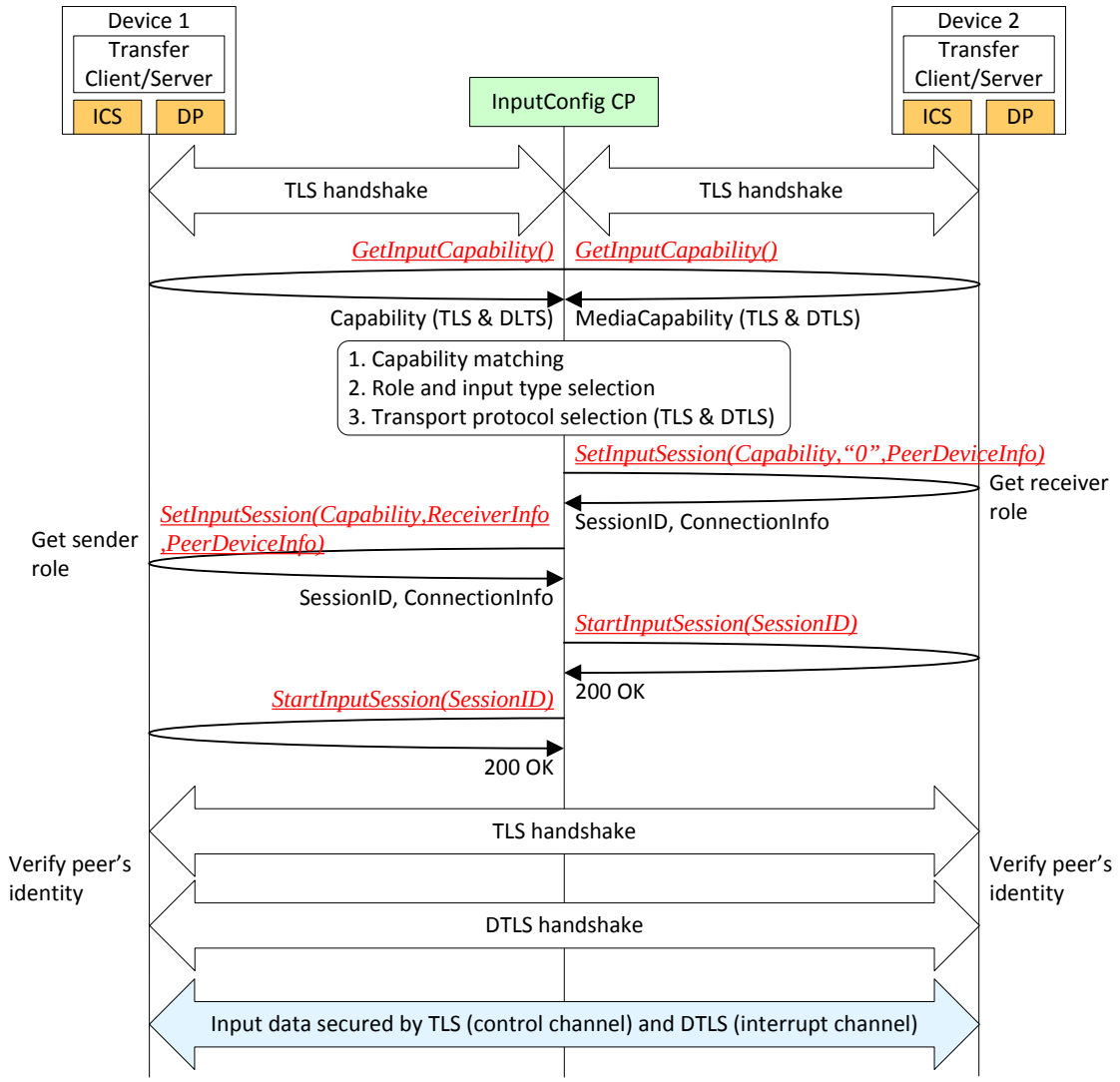
Copyright © 2011 UPnP Forum. All rights reserved.

# 5 Secure Input Sessions

In the *InputConfig* service [ICS], the TLS (Transport Layer Security) and the DTLS (Datagram TLS) are used as the secure transport protocols for an input session. In addition, the input argument *DeviceInfo* of the *SetInputSession()* action contains the <*deviceCertID*> tag, which includes the peer device's (e.g. sender in the context of receiver) Device Identity as defined in the *DeviceProtection* [DP] – *DeviceProtection:1*, UPnP Forum, February 2011.
Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.
Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1*, UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

. When the Input Control Point (ICP) sets this tag, then the device will use this identity to authenticate the peer device while establishing an input session, to make sure that the peer device is an authorized sender/receiver for the input session. The figure and sequence text below illustrates the procedures for setting up the secure input session between the two devices.




Copyright © 2011 UPnP Forum. All rights reserved.

**Figure 6: Secure input session set up by an ICP**

1. Before invoking any actions on input devices, the ICP sets up TLS connections with both the devices in order to deliver the input session configuration parameters in a secure way. The device which implements the *DeviceProtection* service should check the role of ICP to determine if the ICP is authorized to configure the input device.
2. The ICP invokes the *GetInputCapability()* action to discover the input capabilities of a device, and supported secure transport protocols (e.g., TLS and/or DTLS). If devices support both the secure transport protocols, then the ICP can instruct devices to use the TLS and the DTLS to secure both the control and the interrupt channels for the input session.
3. The ICP decides the input session parameters including the security protocols. The ICP delivers these parameters by invoking the *SetInputSession()* action on each device. In order to set up secure input session, the ICP should include the "TLS" and the "DTLS" values in the *SelectedCapabiliy* input argument. Furthermore, the <*deviceCertID*> tag of the *PeerDeviceInfo* input argument should be set to the peer device's Device Identity, which will be used for the authentication.
    - Herein Device Identity is a UUID value derived from the device's certificate as specified in *DeviceProtection* [DP] – *DeviceProtection:1,* UPnP Forum, February 2011. Available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf. Latest version available at: http://www.upnp.org/specs/gw/UPnP-gw-DeviceProtection-v1-Service.pdf.

[CaMS] – *CallManagement:1,* UPnP Forum, March 22, 2011.
Available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service-20110322.pdf.
Latest version available at: http://www.upnp.org/specs/phone/UPnP-phone-CallManagement-v1-Service.pdf.

    - .
4. After receiving the *StartInputSession()* action, the sender device initiates the TLS and DTLS handshakes with the receiver device. The sender and receiver device should calculate the peer's Device Identity value from the exchanged certificate, and verify this value with the previously received value from the ICP in order to authenticate each other. Then the sender and the receiver device negotiates the session crypto parameters (e.g. session keys) for securing (i.e. encrypting and integrity protecting) the input data and the control messages.
    - Note it is up to the device implementation to decide when to perform the authentication on peer device's identity using <*deviceCertID*>. The implementation can authenticate during the TLS/DTLS handshakes and before negotiating the session parameters, or after handshakes when receiving the first input data or the control message from the peer device.
5. When the input session is successfully established, the DTLS tunnel protects the actual input data, and the TLS tunnel protects the control messages.

Copyright © 2011 UPnP Forum. All rights reserved.