



M5T SIP SAFE v4.0 Overview and Description

Copyright © 2005 M5T, Inc.

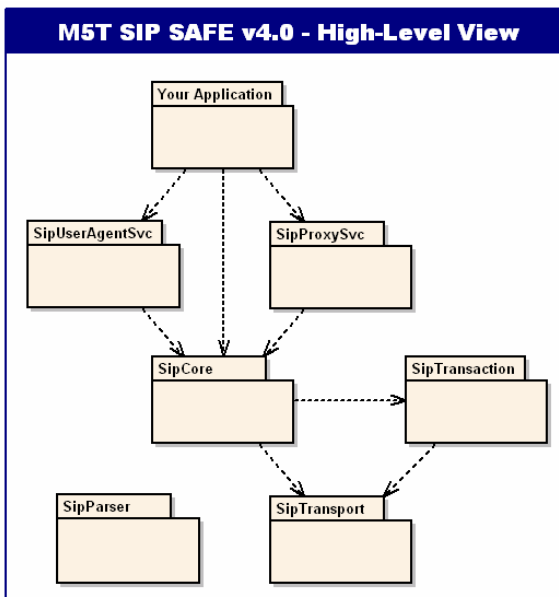
M5T, Inc. reserves all rights to this document as well as to the Intellectual Property of the document and the technology and know-how that it includes and represents.

M5T, Inc. reserves the right to revise this publication and make changes at any time and without the obligation to notify any person and/or entity of such revisions and/or changes.

1. Introduction to M5T SIP SAFE

M5T SIP SAFE is the latest RFC 3261 based C++ SIP stack generation which can be used for creating standard and advanced secure SIP devices, servers and applications. Bearing M5T's security strategy tag "SAFE", it offers built-in security and advanced functionality to help build feature rich yet highly secure SIP products.

M5T SIP SAFE offers much more than other SIP stacks. Where other stacks offer parsers, transport and transaction management layers with some helpers for managing dialogs, M5T SIP SAFE offers all this along with numerous SIP-based modular services that can be used within the provided high level SIP framework.



- **SIP Parser:** Package offering parser/generator and container classes for SIP data.
- **SIP Transport:** Package abstracting the SIP transport and managing connections to other hosts.
- **SIP Transaction:** Package that manages SIP retransmits and detects the retransmission of received packets.
- **SIP Core:** Abstract package offering the basic framework for the use of modular services by applications.
- **SIP User Agent Services:** Package offering a set of modular user agent services that are used by the application.
- **SIP Proxy Services:** Package offering a set of modular SIP proxy oriented services that are used by the application.

M5T SIP SAFE is ideally suited for intelligent SIP devices, small or large, such as the following applications.

- IP Phones
- PC Clients
- CPEs and Gateways
- Mobile, Wireless and Wi-Fi devices
- PDAs
- Systems on a chip
- Applications for collaboration
- Proxy servers
- Redirect servers
- Registrar servers
- Application servers
- Conferencing servers
- ... And any other kind of SIP system.

As with all other products developed from M5T, M5T SIP SAFE is developed over the M5T Framework. The M5T Framework is a suite of tools, algorithms and patterns serving as the foundation for building advanced products. The M5T Framework abstracts the operating system, the network access and simplifies threading for all M5T products. It can also be directly accessed and used as a foundation for portable, multithreaded applications.

2. Key features

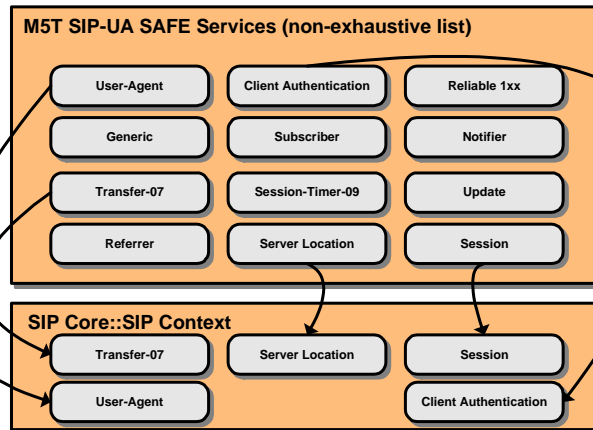
Modular Services

The key differentiator of M5T SIP SAFE is the set of user agent and proxy services it offers. These services are actually modular “service-logic” implementation of specific SIP features that can be enabled per dialog or request. All the provided service logic definitely allow for faster development and better interoperability.

Code Size

M5T SIP SAFE was developed for building intelligent devices. Nowadays we find embedded and server-scale devices that require all this intelligence. M5T SIP SAFE was built for minimizing its footprint and dynamic memory consumption, which is often a critical issue for small and large devices. Moreover, the modular service architecture offered by the stack allows applications to only use the services it requires, without having to pay the price in flash and ram for the services it does not. Developers only need to remove the unnecessary services from the compilation build to have a smaller application.

The example below shows a pool of services that were compiled with an application. When the application chooses to start a new session (voice, video, data, etc), it creates a new “SIP Context” object and loads all the services that it wants to be active for this session. The application can create as many “SIP Context” instances as it requires, each with different services configuration.



Backward Interoperability

The modular approach offered by M5T SIP SAFE can circumvent many interoperability issues, as multiple versions of services are supported at run-time. An application just has to be pre-configured with the version of the service to use or dynamically detect which internet-draft version the other peer is using. The application can then load the proper service on a SIP Context and ensure its backward compatibility with older applications. As an example, an application could choose to use the latest “transfer-07” service when calling a specific destination, and use the older “transfer-02” service when calling another destination that was not yet updated with the latest standards.

Extension Mechanisms

M5T SIP SAFE was built with an extensive knowledge of SIP and its extensions. However, there are still a lot of SIP extensions yet to be developed. For this reason, M5T SIP SAFE allows applications to specify a set of additional SIP headers whenever any SIP message is sent. Moreover, applications can use the generic service for sending and receiving any type of requests and responses.

Security

M5T is renowned for its security offering, and M5T SIP SAFE is no exception. Committed to support future security mechanisms, M5T SIP SAFE already supports TLS as a secure transport for SIP packets, with one way or mutual TLS authentication. Moreover, the standard digest authentication (MD5) and advanced Kerberos authentication are also implemented.

To complement M5T SIP SAFE security features, M5T also offers M5T SRTP SAFE and M5T MIKEY as separate products.

3. Supported Standards

M5T SIP SAFE not only allows applications to implement a wide range of existing SIP standards through its extension mechanisms, it also provides, through its modular service oriented architecture, the actual service logic implementation for an impressive number of standards.

| PROVIDES SERVICE LOGIC FOR | SUPPORTED |
|---|--|
| RFC 1510 - The Kerberos Network Authentication Service (V5) RFC 2246 - TLS RFC 2327 - SDP - Session Description Protocol RFC 2543 - SIP - Session Initiation Protocol RFC 2617 - HTTP Auth - Basic and Digest RFC 2976 - The SIP INFO Method RFC 3261 - SIP - Session Initiation Protocol RFC 3262 - Reliable 1xx Responses RFC 3263 - Locating SIP Servers RFC 3264 - SDP Offer-Answer Model RFC 3265 - SIP Specific Event Notification RFC 3311 - The SIP UPDATE Method RFC 3320 - SigComp (through M5T Sigcomp package) RFC 3326 - The Reason Header Field for SIP RFC 3420 - Internet Media Type message-sipfrag RFC 3428 - SIP Extension for Instant Messaging RFC 3455 - 3GPP P-Headers for SIP RFC 3485 - SIP Static Dictionary for SigComp RFC 3486 - Compressing SIP RFC 3515 - The SIP REFER Method RFC 3842 - Message Summary and MWI Event Package for SIP RFC 3891 - The SIP Replaces Header RFC 3966 - The tel URI for Telephone Numbers RFC 4028 - SIP Session-Timers | RFC 3087 - Control of Service Context using SIP Request-URI RFC 3310 - HTTP Digest Authentication Using Authentication and Key Agreement (AKA) RFC 3312 - Integration of Resource Management and SIP RFC 3313 - Private SIP Extensions for Media Authorization RFC 3323 - A Privacy Mechanism for SIP RFC 3325 - Private Extensions to SIP for Asserted Identity within Trusted Networks RFC 3327 - SIP Extension Header Field for Registering Non-Adjacent Contacts RFC 3329 - Security Mechanism Agreement for SIP RFC 3388 - Grouping of Media Lines in SDP RFC 3398 - ISUP to SIP Mapping RFC 3407 - SDP Simple Capability Declaration RFC 3578 - Mapping of ISUP Overlap Signaling to SIP RFC 3603 - Private SIP Proxy-to-Proxy Extensions for Supporting PacketCable DCS Architecture (Parser) RFC 3605 - RTCP attribute in SDP RFC 3608 - SIP Extension Header for Service Route Discovery During Registration RFC 3680 - Event Package for Registrations RFC 3725 - BCP for 3PCC in SIP (*3gpp) RFC 3840 - Indicating User Agent Capabilities in SIP RFC 3841 - Caller Preferences for SIP RFC 3856 - A Presence Event Package for SIP RFC 3857 - A Watcher Information Event Template-Package for SIP RFC 3903 - SIP Extension for Event State Publication RFC 3911 - The SIP Join Header |

The following modular services are offered in the stack to implement the provided service logic:

| User Agent Services | Proxy Services |
|--|--|
| ISipDigestClientAuthSvc ISipGenericSvc ISipGlareSvc ISipKerberosClientAuthSvc ISipMwiSvc ISipNotifierSvc ISipRedirectionSvc ISipRefereeSvc ISipReferrerSvc ISipRegistrationSvc ISipReliableProvisionalResponseSvc ISipReplacesSvc ISipSessionSvc ISipSessionTimerSvc09 ISipSubscriberSvc ISipTransferSvc07 ISipUpdateSvc ISipUserAgentSvc | ISipStatelessProxySvc ISipTransactionStatefulProxySvc Common Services ISipConnectionBlacklistSvc ISipCoreOutputControllingSvc ISipPersistentConnectionSvc ISipServerLocationSvc ISipSigComplmsSvc ISipStatelessDigestServerAuthSvc ISipSymmetricUdpSvc |

4. Requirements

M5T SIP SAFE requires an ANSI C++ Compiler with support of templates, placement new and placement delete operators, and support of the "mutable" keyword.

Combined with the M5T Framework, the stack is already ported to the following operating systems:

- Linux
- VxWorks
- Windows
- WindowsCE

Additional operating systems can easily be supported.