DEVELOPMENT APPROACHES FOR AN INTERNATIONAL TACTICAL RADIO CRYPTOGRAPHIC API

Antonio Martin (SCA Technica, Inc., Nashua, NH; tony.martin@scatechnica.com)
Timothy R. Newman, Ph.D. (Virgina Tech, Blacksburg, VA; trnewman@vt.edu)
David Murotake, Ph.D. (SCA Technica, Inc. Nashua, NH; dmurotak@scatechnica.com)

ABSTRACT

Currently, there are no internationally available application programming interfaces (APIs) for the embedded cryptographic module of modern tactical radios which comply with the Joint Tactical Radio System (JTRS) Software Communications Architecture (SCA). We survey relevant, internationally released documents including recommendations from the SDR Forum Security Working Group, publicly released information on the Common Interfaces to Cryptographic Modules (CICM) project and SCA specifications including the SCA 3.0 Security Supplement. These and other documents are examined as the possible basis for a draft Cryptographic API employing common interfaces for an embedded cryptographic module usable by an International Tactical Radio (ITR) and Waveform Development Environment (WDE). The draft Cryptographic API can be used as the basis for a future recommendation by the SDR Forum Security Working Group.

1. INTRODUCTION

A goal of the SCA is to enhance the portability of waveforms across radios and to leverage commercial standards to reduce development costs. The JTRS has responsibility for the SCA standard and supporting service, device and hardware abstraction APIs. [1] To this end, the JTRS recently released thirteen APIs providing waveform interfaces for the following abstracted types:

AudioPortDevice API – Provides an interface to control and data for audio port devices (push to talk) [2]
DeviceIO API – Interfaces to enable and set RTS and CTS messages [3]
DeviceIOControl API – Interfaces to enable RTS, CTS and set only RTS messages [4]
DeviceIOSignals API – Provides interface to start and stop packet flows and set CTS. [5]
DeviceMessageControl API – Allows for setting transmit and receive active and terminating transmit streams. [6]
DevicePacket API – Controls settings for payload size, priorities and flows for data. [7]
DevicePacketSignals API – Receives settings for payload size, priorities and flows for data. [8]
EthernetDevice API – Interface for controlling and abstracting Ethernet devices. [9]
JTRS CORBA Types – Defines set CORBA types. [10]
Packet API – Interfaces to control and push packets to devices. [11]
SerialPortDevice API – Abstraction for interfacing to serial port devices. [12]
VocoderService API – Abstraction for vocoder devices. [13]
ModemHardwareAbstractionLayer API – An abstraction layer for DSPs, FPGAs and other non-GPP devices. [14]

While the JTRS supports an API providing cryptographic and transmission security services for waveforms, it has not been publicly released. This provides a stumbling block for commercial and international alied development of SCA waveforms and components.

A red black design requires a radio to have three distinct areas: a Red and Black processing elements separated by a cryptographic engine. The purpose is to force isolation between unprotected, unencrypted sensitive data and a transmitting unit helping to prevent accidental (or not) leaks of unprotected information. Each crypto unit is proprietary and has difference ports and messaging; thus

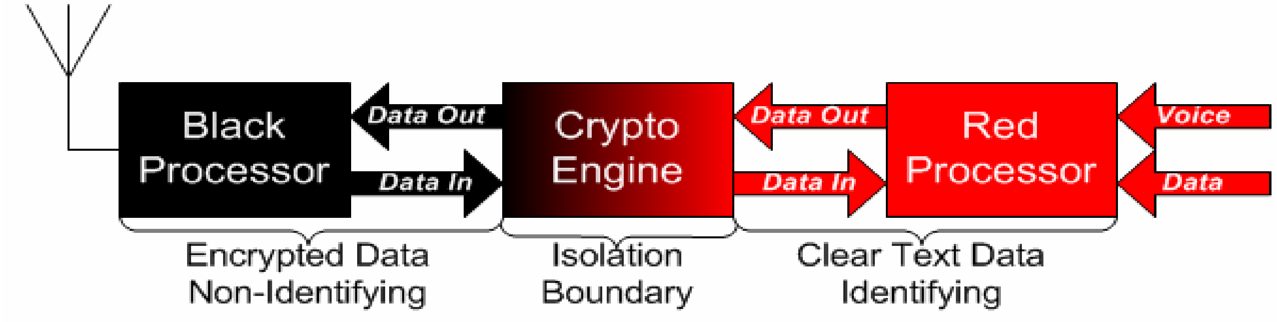

Figure 1: Red Black radio isolation

is the need to abstract the interfaces from the portable waveform.

2. SECURITY OVERVIEW

System security spans well beyond the capabilities or function of an encryption engine. It is a total solution spanning from a device's hardware and software to transmitted energy and information and the network formed from intercommunication. The most commonly addressed layers are listed below:

- Communication Security
- Computer Security
- Information Security
- Network Security
- Signal Security
- Transmission Security

The development approaches for the international API described in this paper focus on the COMSEC and TRANSEC services.

Communication security (COMSEC) is the protection resulting from all measures designed to deny unauthorized access to information of value derived from the possession and study of telecommunications, or to mislead unauthorized persons in their interpretation of the results of such possession and study. [15] For example, if an unauthorized person were to capture data from a wireless transmission generated using a COMSEC service, no information would be able to be recovered from the transmission due to encryption.

Transmission security (TRANSEC) is the component of communication security resulting from measures designed to protect transmissions from interception and exploitation by means other than cryptanalysis. [15] The goal is to deny unauthorized users the opportunity to intercept or exploit the transmission. One common way to provide this service is through frequency hopping. U.S. and NATO TRANSEC-enabled radios include SICGARS [16] and HAVE QUICK [17].

While the other security services are important to complete system security, the radio cryptographic APIs for SDR architectures do not to focus on them. Computer security (COMPUSEC) is the protection of the computational platform and software and information being processed and stored from exploitation. [15] This entails multiple layers including physical locks to protect the hardware and by using a secure operating system such as SELinux [18]. Network security (NETSEC) is the protection of the integrity of the network and network services. This includes the protection of critical network entities such as routers, gateways, and firewalls.

Signal security (SIGSEC) is study of friendly system signals to monitor for exploitable and accidental electromagnetic spectrum emissions. Information security service (INFOSEC) is the primary, over arching security classification, the protection of information from exploitation, compromise and authenticity. It leverages all other security functions and services. [15]

3. EXISTING CRYPTOGRAPHIC SPECIFICATIONS

There are at least three existing API standards supporting classified crypto units. While several commercial APIs exist, they at the least do not support security domain separation necessary for processing classified information.

3.1 Federal Information Processing Standards (FIPS)

Any abstraction for a cryptographic device must first understand how they behave and are used. The FIPS PUB 140-2 Security Requirements for Cryptographic Modules provides excellent basis of information for cryptographic units. It defines a unit having four required logical interfaces: [19]

- A data input interface for accepting information to be transformed.
- A data output interface where transformed information shall be sent.
- A control input interface for setting up and modifying transformation functions.
- A status output interface to provide system feedback.
- Optional is a power port in the case a unit has an internal battery supply.

While the FIPS 140-2 is designed for unclassified cryptographic units, it is an example of openly published information useful for future Cryptographic API development.

3.2 Common Interfaces to Cryptographic Modules (CICM)

Common Interface Cryptographic Module (CICM) is an API for a wide class of crypto units and services being developed at MITRE under the Air Force's Cryptographic Modernization program. CICM is designed to be a generic interface supporting not just streaming data but also stored information protection and verification. While similar, an SCA enabled tactical radio would only leverage a subset because of its limited scope. CICM provides interfaces supporting: Module, Algorithm, Policy, Channel, Data, Key and Trust anchor management. The CICM approach is to let the cryptographic module enforce behavior while the

API acts as a bridge, passing messages between the application and the hardware.

CICM's goal is to develop an unclassified specification without any restrictions and ownership passed to an international standards organization. Currently, CICM has been implemented and is working on the Advanced Beyond Line-of-Sight Terminals (FAB-T) program. [20, 21, 22]

3.3 SCA Security Supplement

The SCA Security Supplement was released with the SCA 2.2 and carried over to version 3.0; it was not included in version 2.2.2. It is a publicly available document downloadable from the JTRS's website (http://sca.jpeojtrs.mil/downloads.asp?ID=3.0).

The 3.0 version is comprised of the following three documents:

– SCA Security Supplement v3.0
– Security API Service Definitions
– Function Security Requirements

SCA Security Supplement v3.0 – This is the primary document describing the functional requirements and system behavior for an SCA enabled radio. It describes the functional architecture and behavior of a crypto subsystem and the various security boundaries. A form of Bypass is described, allowing for trusted information to flow across the red black boundary. Also addressed are the functional separations of the Red, Crypto and Black processing engines. [23]

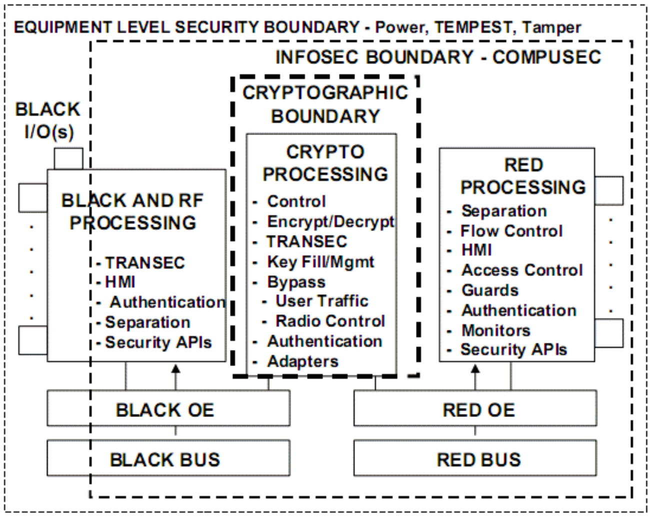

Figure 2: Cryptographic sub-system boundaries [23]

Attachment 1 Security Application Program Interface Service Definitions specifies, in detail, the API for security services for SCA enabled radios with IDL. [24] Appendix A – Functional Security Requirements provides a matrix of all functional requirements for the security APIs. [25]

The 3.0 version is comprised of twelve security service groups as shown in Fig. 3. Security provides a system zeroize functionality, Fill provides fill information for the cryptographic unit, Algorithm enables reprogramming and management of algorithms, Certificate for digital signatures and key exchange needed for some algorithms, Crypto for COMSEC channel management and control, Key for persistent keys needing storage, TRANSEC for keystream information, Policy enforces security requirements, Integrity and Authentication for verification of data, Alarm for audit recording and alarms, Time and GPS [24].

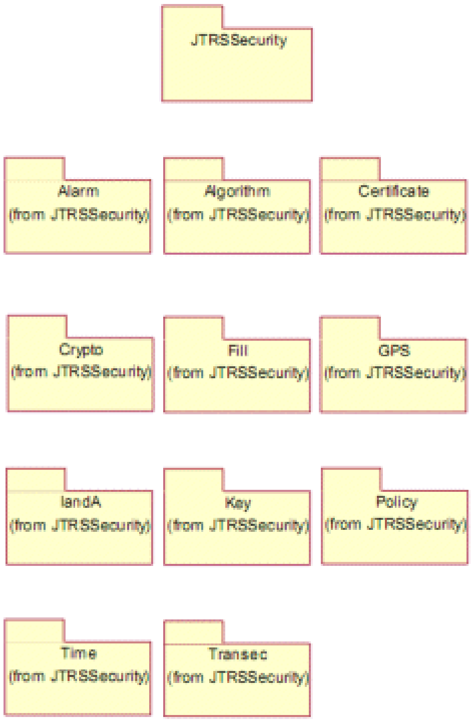
JTRSSecurity
Alarm (from JTRSSecurity)
Algorithm (from JTRSSecurity)
Certificate (from JTRSSecurity)
Crypto (from JTRSSecurity)
Fill (from JTRSSecurity)
GPS (from JTRSSecurity)
IandA (from JTRSSecurity)
Key (from JTRSSecurity)
Policy (from JTRSSecurity)
Time (from JTRSSecurity)
Transec (from JTRSSecurity)

Figure 3: SCA 3.0 Security Services API [22]

In a 2005 presentation to the Object Management Group (OMG), Frisina *et al.* presented the following set of concerns with the SCA Security Supplement API as it related to the Ground Mobile Radio (GMR) [26]:

– *Defines separate abstractions for COMSEC and TRANSEC channels*
– *Unclear how the multiple keys permitted within a COMSEC channel are utilized*
– *Does not provide for multiple keys within a TRANSEC channel*

- *Permits multiple modes of operation within a COMSEC channel however significant overhead to switch between modes in real time*
- *Does not provide for specifying mode of operation within a TRANSEC channel.*
- *Does not provide for management of cryptographic state within a channel or between COMSEC and TRANSEC channels*
- *Does not provide for dynamic key exchange*
- *Does not provide for Over the Air Re-key (OTAR) and Over The Air Transfer (OTAT)*

The SCA Security Supplement documents are extensive, offering a wealth of information and can be used as a as basis for a Cryptographic API for International Tactical and National Security Radios.

3.4 Radio Security Service (RSS)

Introduced as the security services for the Ground Mobile Radio (GMR), the Radio Security Services (RSS) provides components derived from the SCA security API. Several changes were needed to adapt the API to the needs of GMR waveform applications. Several parts of the security API were not used, some were modified and several new functions were added in order to fill in the missing functionality.

The initial development of the RSS began with the SCA Security Supplement API as the foundation. Several factors played into the practical realization of the API causing the development of the RSS to diverge from the foundation:

- Portability of the cryptographic devices
- Functional security
- Performance of the channel transformations
- Legacy waveform requirements
- Future waveform requirements

The RSS attempts to address the needs of each of these factors by removing unneeded services from the SCA security API, such as the alarm and time service, adding new services for increased functionality, such as the audit service group, user authentication service, and services to support Over The Air (OTA) re-key and OTA transfer. In addition to defining a unified cryptographic channel with separate abstractions for COMSEC and TRANSEC configurations, significant extensions to existing abstraction of cryptographic channels are provided to address key elements of missing functionality.

- Enabling the use of multiple keys and modes
- Dynamic key generation
- Management of the cryptographic state

Although not a publicly available specification, the RSS is intended to provide JTRS radios with a more secure and functional cryptographic device interface. Building from the original SCA 3.0 Security Supplement, the RSS is a more developed and functional set of cryptographic specifications. [26]

4. USE CASES AND REQUIREMENTS FOR AN INTERNATIONAL TACTICAL RADIO CRYPTO API

The ITR API goal is to develop openly available crypto APIs focused on COMSEC and TRANSEC functions and is seen as a two phase approach. The first is to develop a set of use cases based on input and prior documentation. The next is to develop platform independent model and behavioral descriptions for the interfaces. While these APIs might not match the RSS exactly, functional interface limitations should result in an API broadly similar to (and interoperable with) the US JTRS RSS standard.

4.1. International Tactical Radio Special Interest Group (ITR SIG).

The ITR SIG is newly established. Its purpose is to bring together users, regulators, developers, manufacturers and buyers of tactical radios (used by Defense and Homeland Security operators around the world), and collaboratively develop business case studies, analyze user requirements, and develop use cases for SDR based international tactical radios. Although not strictly JTRS oriented, many of the ITR SIG members come from national administrations or agencies which have signed MOUs, bilateral treaties, or multilateral treaties with the USA establishing the JTRS SCA as an interoperability standard. Thus, a use case for a Crypto API interoperable with the JTRS SCA is for the purpose of interoperability with US forces in joint coalition operations world-wide.

Unfortunately, the latest versions of the JTRS SCA 2.2.2 Radio Security Services (RSS) API are not exportable outside of the US and are currently restricted to a small group of approved JTRS authorized developers. The ITR SIG has identified an acceptable alternative approach, basing its approach on the recently published SCA 3.0 Security Supplement mentioned previously, in addition to other resources including recently voted recommendations from the SDR Forum's Security Working Group. The ITR SIG, working together with the Security and API Working Groups, plan on a three-step development plan shown in Figure 4.1.1 below.

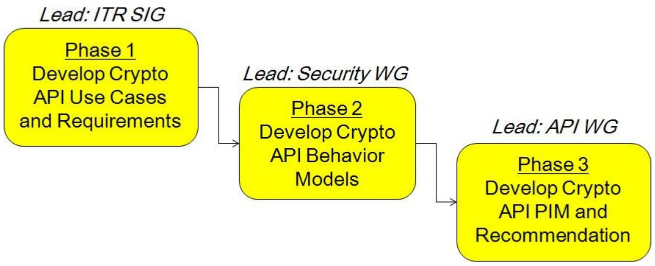
Lead: ITR SIG
Phase 1
Develop Crypto API Use Cases and Requirements
Lead: Security WG
Phase 2
Develop Crypto API Behavior Models
Lead: API WG
Phase 3
Develop Crypto API PIM and Recommendation

Figure 4: The Crypto API project should be conducted in three stages by the SDR Forum, with leadership shared between the ITR SIG, Security WG, and API WG.

4.2. ITR Crypto API Use Cases

At least two use cases exist for the ITR Crypto API. These include:

- Use Case #1. JTRS SCA Compatible Red/Black Tactical Radio. This radio is capable of up to Type 1 embedded encryption. Its requirements and API will be based on the published SCA 3.0 Security Supplement and other documents. Development of optional TETRA and APCO-25 public safety waveforms for SCA compliant radios (e.g. Harris Falcon III) make SCA compliant radios attractive to US and European homeland security operators with a requirement to interoperate with defense units equipped with SDR tactical radios.

- Use Case #2. Commercial Off the Shelf (COTS) or COTS-based Radio Sets for Tactical and Homeland Security. This radio employs non-Type 1 embedded encryption, and may include mobile handsets compatible with commercial standards such as GSM, 3GPP1, 3GPP2, WIFI and WIMAX as well as future Cognitive Radio or Dynamic Spectrum Access (DYSPAN).

4.3. Draft High Level Recommended Requirements

- R1. The Crypto API shall have a Platform Independent Model (PIM) compatible with the Object Management Group (OMG) Model Driven Architecture (MDA) [27] and the OMG SWRadio PIM [28].
- R2. The Crypto API shall be an Open Source API freely accessible to the international community and published as a voted Recommendation by one or more international industry group(s) such as the SDR Forum.
- R3. The Crypto API PIM shall support the requirements and behaviors for multiple Use Cases including Use Cases #1 and #2.
- R4. Each use case will be represented by a profile.

5. CONCLUSION

The JTRS SCA is an international standard adopted by many countries. A continuing cause for concern is the lack of standardized APIs to assist in waveform portability. Although a number of important APIs have been released, a key API - the Radio Security Services (RSS) API for the current SCA - remains ITAR restricted and unavailable for international scrutiny.

The purpose of an internationally available Crypto API is to assist in the portability of waveforms across platforms which incorporate an embedded cryptographic module, including Type 1 cryptographic modules embedded in Red/Black isolated radio sets. The development of an open, international crypto API for tactical radios will allow international partners to develop interoperable waveform components for use in coalition operations. A set interface to known devices allows waveform components to be more easily ported to new hardware platforms.

The SCA 3.0 Security Supplement provides an excellent basis for future work. Although the SCA 3.0 itself is no longer a supported baseline by the US JTRS program, its Security Supplement is the latest before the JTRS program chose to standardize on SCA v2.2.2. SCA 3.0 security supplement forms a good, open-specification basis for an international Crypto API. A future, unrestricted release of either CICM or the JTRS RSS API could also help in the development of an internationally accepted interface to cryptographic modules for classified information processing in tactical radios. The SDR Forum's ITR SIG is in a unique position to develop / start such a standard for future waveform development.

A Security API provides but a subset of the necessary protection for a radio. "Encryption is not enough" holds true when protecting reconfigurable resource elements. [29, 30]

6. REFERENCES

[1] B. Salisbury, V. Popik, “JPEO JTRS Overview to OMG” SBC Workshop, Object Management Group, August 2005
[2] “Joint Tactical Radio System (JTRS) Standard Audio Port Device IO Application Program Interface (API)” Version: 1.3.2, April 2008
[3] “Joint Tactical Radio System (JTRS) Standard Device IO Application Program Interface (API)” Version: 1.0.1, March 2007
[4] “Joint Tactical Radio System (JTRS) Standard Device IO Control Application Program Interface (API)” Version: 1.1.1, March 2007

[5] "Joint Tactical Radio System (JTRS) Standard Device IO Signals Application Program Interface (API)" Version: 1.1.1, March 2007

[6] "Joint Tactical Radio System (JTRS) Standard Device Message Control Application Program Interface (API)" Version: 1.1.1 29 March 2007

[7] "Joint Tactical Radio System (JTRS) Standard Device Packet Application Program Interface (API)" Version: 1.1.1 29 March 2007

[8] "Joint Tactical Radio System (JTRS) Standard Device Packet Signals Application Program Interface (API)" Version: 1.2.2, April 2008

[9] "Joint Tactical Radio System (JTRS) Standard Ethernet Device Application Program Interface (API)" Version: 1.1.1 , March 2007

[10] "Joint Tactical Radio System (JTRS) Standard JTRS CORBA Types" Version: 1.0.2, April 2008

[11] "Joint Tactical Radio System (JTRS) Standard Packet Application Program Interface (API)" Version: 2.0.2, April 2008

[12] "Joint Tactical Radio System (JTRS) Standard Serial Port Device Application Program Interface (API)" Version: 2.0.2, April 2008

[13] "Joint Tactical Radio System (JTRS) Standard Vocoder Service Application Program Interface (API)" Version: 1.1.1.1, August 2007

[14] "Joint Tactical Radio System (JTRS) Standard Modem Hardware Abstraction Layer Application Program Interface (API)" Version: 2.11.1, May 2007

[15] "National Information System Security (INFOSEC) Glossary" NSTISSI No. 4009, National Security Agency, September 2000

[16] "Single Channel Ground and Airborne Radio System (SINCGARS)" Federation of American Studies, March 1999 http://www.fas.org/man/dod-101/sys/land/sincgars.htm

[17] "AN/ARC-164 HAVE QUICK II" Federation of American Studies, January 1999 http://www.fas.org/man/dod-101/sys/ac/equip/an-arc-164.htm

[18] Security Enhanced Linux, http://www.nsa.gov/selinux/

[19] "Security Requirements for Cryptographic Modules" Federal Information Processing standards Publication, FIPS PUB 140-2, Information Technology Laboratory, May 2001

[20] D. Lanz, "Common Interface to Cryptographic Modules" presentation slides, Software Defined Radio Forum – Rome, April 2008

[21] S. Cardman, "Common Interface to Cryptographic Modules Analysis of Security APIs Workshop, June 2008 http://www.lsv.ens-cachan.fr/~steel/asa2/Cardman-09-CICM.pdf

[22] J. Keller, "Air Force Making Process on Standard Cryptographic Modules for Information Security" Military & Aerospace Electronics, April 2008

[23] "Security Supplement to the Software Communications Architecture Specification" Joint Tactical Radio System, Joint Program Office, JTRS-5000 SEC V3.0, August 2004

[24] "Attachment 1 Security Application Program Interface Service Definition" Joint Tactical Radio System, Joint Program Office, JTRS-5000 SEC V3.0, August 2004

[25] "APPENDIX A Functional Security Requirements for JTRS" Joint Tactical Radio System, Joint Program Office, JTRS-5000 SEC V3.0, August 2004

[26] J. Frisina, L. Monahan, D. Retotar, "Security API Developments in JTRS Cluster 1" OMG Software Based Communications Working Group – San Diego, August 2005

[27] "MDA® Specifications" Object Management Group November 2007 http://www.omg.org/mda/specs.htm

[28] "PIM and PSM for SWRADIO Draft Revised Submission – swradio/2004-01-01" OMG Specification. January 2004, www.omg.org/docs/swradio/04-01-01.pdf

[29] D. Murotake, A. Martin, "System Threat Analysis for High Assurance Software Defined Radios," SDR Forum, November 2004

[30] W. Scott, A. Houle, A. Martin, "Information Assurance Issues for an SDR Operating in a MANET Network," SDR Forum, November 2006

Copyright Transfer Agreement: The following Copyright Transfer Agreement must be included on the cover sheet for the paper (either email or fax)—not on the paper itself.

"The authors represent that the work is original and they are the author or authors of the work, except for material quoted and referenced as text passages. Authors acknowledge that they are willing to transfer the copyright of the abstract and the completed paper to the SDR Forum for purposes of publication in the SDR Forum Conference Proceedings, on associated CD ROMS, on SDR Forum Web pages, and compilations and derivative works related to this conference, should the paper be accepted for the conference. Authors are permitted to reproduce their work, and to reuse material in whole or in part from their work; for derivative works, however, such authors may not grant third party requests for reprints or republishing."

Government employees whose work is not subject to copyright should so certify. For work performed under a U.S. Government contract, the U.S. Government has royalty-free permission to reproduce the author's work for official U.S. Government purposes.