CPA SECURITY CHARACTERISTIC

SECURE REAL-TIME COMMUNICATIONS CLIENT

Version 2.1

About this document

This document describes the features, testing and deployment requirements necessary to meet CPA certification for Secure real-time communications client security products. It is intended for vendors, system architects, developers, evaluation and technical staff operating within the security arena.

- Section 1 is suitable for all readers. It outlines the purpose of the security product and defines the scope of the Security Characteristic.
- Section 2 and Section 3 describe the specific mitigations required to prevent or hinder attacks for this product. Some technical knowledge is assumed.
- For more information about CPA certification, refer to The Process for Performing CPA Foundation Grade Evaluations[1].

Document history

The CPA Authority may review, amend, update, replace or issue new Scheme Documents as may be required from time to time. Soft copy location: DiscoverID 41021167

Version	Date	Description
2.0	February 2013	First release with new template
2.1	December 2014	Amendments to allow the Security Characteristic to cover more applications, reflected in updated document title

This document is derived from the following SC Maps.

SC Map	Map version
Secure real-time communications client	2.1.2
Common Libraries	2.1.4
Crypt Libraries	2.1.4

Contact CESG

This document is authorised by: Technical Director (Assurance), CESG.

For queries about this document please contact:

CPA Administration Team
CESG, Hubble Road
Cheltenham
Gloucestershire
GL51 0EX, UK

Email: cpa@cesg.gsi.gov.uk
Tel: +44 (0)1242 221 491

[1] www.cesg.gov.uk/servicecatalogue/CPA

Contents

1.1 Introduction

This document is a CPA Security Characteristic. It describes requirements for assured Secure real-time communications client products for evaluation and certification under CESG's Commercial Product Assurance (CPA) scheme.

1.2 Product description

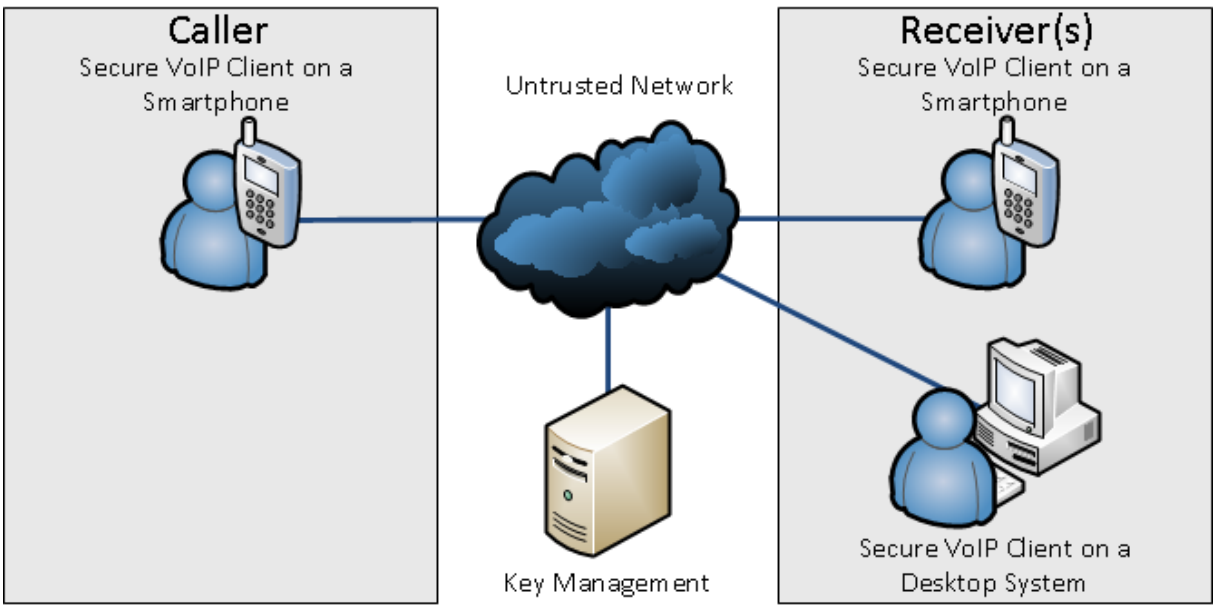
Secure real-time communications clients are used to send and receive real-time media over an untrusted network. The client provides an encrypted and mutually authenticated channel for real-time communications to be made between mobile or fixed devices.

Secure real-time communications clients considered in this document are specifically software applications installed on a general-purpose operating system such as a Desktop Computer or Smartphone.

For clarity, the two types of real time communications within the scope of this Security Characteristic are Voice over IP (VoIP) and secure video conferencing, both using Secure Real-time Transport protocol (SRTP).

1.3 Typical use cases

The Secure real-time communications client will typically be used to hold sensitive voice or video conversations in disparate locations between two or more parties connected via untrusted networks. The diagram below shows an example of a caller using the Secure real-time communications client to communicate with multiple parties over an untrusted network.



The host platform would normally be a personal-issue, single-user device or a shared device where user authentication is handled by the host platform. These devices would typically form part of a remote working solution, but could also be part of an on-site enterprise infrastructure.

1.4 Expected operating environment

The Secure real-time communications client must run on devices that are configured according to CESG End User Devices Guidance for that device [e], and are accredited to protect data at the highest level of sensitivity of communications handled by the device.

The application is likely to be run on devices which are attached to untrusted public networks, and will be able to communicate securely over those networks.

The application might be used in a physical location where it is at risk of being overlooked or overheard, but does not attempt to mitigate this risk.

The Secure real-time communications client may utilise external signalling services (such as SIP) to facilitate call connections. As these services do not need to be trusted to ensure confidentiality and integrity, their configuration and use is out of scope of this document. However, these stacks must be robust, and will be subject to testing as part of the holistic product assessment.

Secure real-time communications clients may also be used to make authenticated communications over trusted networks.

1.5 Compatibility

Secure real-time communications clients considered in the context of this document are specifically software applications installed on a general-purpose operating system. The operating system may provide several of the security functions of the product. Dedicated hardware appliances are out of the scope of this document.

The client application should be compatible with modern and up to date operating systems. However there is no requirement to support specific or multiple operating systems.

1.6 Interoperability

Different client products conforming to this Security Characteristic are not currently required to interoperate. Such products are however required to support the MIKEY-SAKKE Streaming Media Profile as a first step towards the long term goal of interoperability.

1.6.1 MIKEY-SAKKE Streaming Media Profile

The MIKEY-SAKKE Streaming Media Profile consists of the following protocols, used with the cryptographic parameter configuration detailed in [d].

Component	Protocol
Media Protocol	SRTP – RFC 3711 [b]
Key Transport	MIKEY-SAKKE – RFC 6509 [c]

1.6.2 Key Management Server

Interactions between Secure real-time communications clients and Key Management servers are not standardised at this time. However, requirements describing the use of key management servers are given in this document at a high level, and any protocols used by the product must demonstrate the specified behaviours.

1.7 Additional threat information

This Security Characteristic does not address the threat of unauthorised interception of the raw communications media at its source or destination (e.g. VoIP conversation being overheard). Additionally, the host platform is assumed to be free of malware – any risk mitigations may not be effective if the host platform has already been compromised.

1.8 High level functional components

The following diagram illustrates the various high level functional components within this product. Components shown with an asterisk (*) represent those relating to specific mitigations listed in Section 3. These are used to structure the Security Characteristic, and to give context to each mitigation.

Figure 1: Functional components of a Secure real-time communications client product

The functional components in Figure 1 are described as follows.

- **Management*** - Controls the client's settings and security behaviours.
- **Key Management*** - Determines how the keys are managed by the software client, and determines the methods used to key the secure calls.
- **Encryption*** - Methods used to encrypt communications and credential data on the network.
- **Session Handling*** - Manages connections between the client and other end points. Also deals with issues around concurrent communication sessions on other applications on the platform.

1.9 Future enhancements

CESG welcomes feedback and suggestions on possible enhancements to this Security Characteristic.

Likely future enhancements to this security characteristic include:

- Support for dedicated hardware appliances providing a real-time communications facility;
- Further definition of interoperability requirements between clients.

Related future Security Characteristics which may be developed include:

- A key-management server

2.1 Requirement categories

All CPA Security Characteristics contain a list of mitigations that describe the specific measures required to prevent or hinder attacks. The mitigations are grouped into three requirement categories; design, verification and deployment, and appear in section 3 of this document in that order.

- **Development mitigations** (indicated by the **DEV** prefix) are measures integrated into the development of the product during its implementation. Development mitigations are checked by an evaluation team during a CPA evaluation.
- **Verification mitigations** (indicated by the **VER** prefix) are specific measures that an evaluator must test (or observe) during a CPA evaluation.
- **Deployment mitigations** (indicated by the **DEP** prefix) are specific measures that describe the deployment and operational control of the product. These are used by system administrators and users to ensure the product is securely deployed and used in practice, and form the basis of the Security Operating Procedures which are produced as part of the CPA evaluation.

Within each of the above categories, the mitigations are further grouped into the functional areas to which they relate (as outlined in the High level functional components diagram). The functional area for a designated group of mitigations is prefixed by double chevron characters ('>>').

For example, mitigations within a section that begins:

Development>>Management

- concern **Development** mitigations relating to the Management functional area of the product.

Note: Mitigations that apply to the **whole** product (rather than a functional area within it) are listed at the start of each section. These sections do **not** contain double chevron characters.

2.2 Understanding mitigations

Each of the mitigations listed in Section 3 of this document contain the following elements:

- The name of the mitigation. This will include a mitigation prefix (**DEV**, **VER** or **DEP**) and a unique reference number.
- A description of the threat (or threats) that the mitigation is designed to prevent or hinder. Threats are formatted in *italic text*.
- The explicit requirement (or group of requirements) that *must* be carried out. Requirements for foundation grade are formatted in **green text**.
- In addition, certain mitigations may also contain additional explanatory text to clarify each of the foundation requirements, as illustrated in the following diagram.

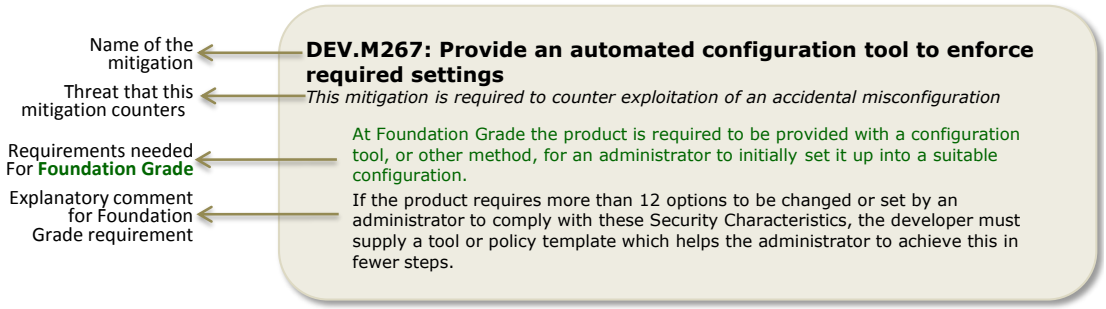


Figure 2: Components of a typical mitigation

This section lists the Development, Verification and Deployment mitigations for the Secure real-time communications client Security Characteristic. For a summary of the changed mitigations in this version, please refer to Appendix A.

3.1 Development mitigations

DEV.M41: Crash reporting

This mitigation is required to counter exploitation of a software implementation/logic error

At Foundation Grade the product **is required to** ensure crashes are logged.

Where it is possible that sensitive data may end up in the crash data, this must be handled as red data and must only be available to an administrator. Crash data from both the product and the underlying operating system must be considered.

DEV.M42: Heap hardening

This mitigation is required to counter exploitation of a software implementation/logic error

At Foundation Grade the product **should** use the memory management provided by the operating system. Products should not implement their own heap.

DEV.M43: Stack protection

This mitigation is required to counter exploitation of a software implementation/logic error

At Foundation Grade the product **is required to** be compiled with support for stack protection including all libraries, where the tool chain supports it.

If more recent versions of the tool chain support it for the target platform then they should be used in preference to a legacy tool chain.

DEV.M159: Update product

This mitigation is required to counter exploitation of a software implementation/logic error

At Foundation Grade the product **should** support the use of software updates.

DEV.M321: Data Execution Prevention

This mitigation is required to counter exploitation of a software implementation/logic error

At Foundation Grade the product **is required to** support Data Execution Prevention (DEP) when enabled on its hosting platform and must not opt out of DEP.

If the product is to be exclusively deployed on a platform that does not support either Software DEP or Hardware-enforced DEP, there is no requirement for DEP compatibility.

DEV.M340: Address Space Layout Randomisation

This mitigation is required to counter exploitation of a software implementation/logic error

At Foundation Grade the product **is required to** be compiled with full support for ASLR, including all libraries used.

If the product is to be exclusively deployed on an operating system that does not support ASLR, there is no requirement for ASLR compatibility.

Note: ASLR may be disabled for specific aspects of the product, provided there is justification of why this is required.

DEV.M351: Erase sensitive data after use

This mitigation is required to counter reading sensitive data in memory

At Foundation Grade the product **is required to** overwrite temporary variables containing sensitive information (such as cryptographic key material) as soon as they are no longer required.

The developer should describe any general approaches taken to help ensure that this requirement is met throughout the product.

DEV.M355: Secure software delivery

This mitigation is required to counter installing compromised software using the update process

At Foundation Grade the product **should** be distributed via a cryptographically protected mechanism, such that the authenticity of software can be ensured.

DEV.M814: Sensitive data storage

This mitigation is required to counter extraction of sensitive data held on the device

At Foundation Grade the product **is required to** store sensitive data using encrypted data protection functions of the host platform.

DEV.M822: Secure host configuration

This mitigation is required to counter exploitation of vulnerabilities caused by client misuse of host platform features

At Foundation Grade the product **is required to** function correctly when the host platform is configured using the relevant CESG End User Devices Security Guidance for that platform.

This guidance will usually be CESG Security Procedures for the host platform.

DEV.M823: Minimum application permissions

This mitigation is required to counter achieving access to host services by exploiting client accesses

At Foundation Grade the product **is required to** request only sandbox permissions necessary for the application to perform its function (if applicable).

DEV.1 - Development >> Session Handling

DEV.1.M809: Mixed classification state prevention

This mitigation is required to counter eavesdropping on a secure communications channel made while an insecure communications channel is active

At Foundation Grade the product **is required to** attempt to prohibit concurrent secure and non-secure communications with any applications on the platform.

DEV.1.M957: Communications security display

This mitigation is required to counter interception of sensitive data accidentally sent unencrypted

At Foundation Grade the product **is required to** identify to the user before the media is received if the media is encrypted, if the other party is authenticated, who the other party purports to be and which entity has attested to their identity.

DEV.2 - Development >> Encryption

DEV.2.M134: State raw entropy requirements

This mitigation is required to counter prediction of randomly generated values due to a weak entropy source

At Foundation Grade the product **is required to** have clearly defined entropy requirements for all operational random number generation.

The developer must state how much raw entropy is required from product's entropy source (i.e. used to reseed the PRNG), based on analysis of all random numbers used in the product, including any generated keys.

At this grade, the amount of raw entropy required is likely to be at least:

- 256 bits for elliptic curve-based asymmetric mechanisms
- 128 bits for symmetric cryptographic mechanisms.

DEV.2.M140: Smooth output of entropy source with approved PRNG

This mitigation is required to counter prediction of randomly generated values due to a weak entropy source

This mitigation is required to counter prediction of randomly generated values due to insufficient raw entropy reaching the PRNG

At Foundation Grade the product **is required to** ensure that all random data used originates from a PRNG that is seeded by an approved entropy source.

The PRNG must also be reseeded sufficiently frequently to avoid raw entropy input being overused.

The PRNG and its reseeding mechanism, which could in certain cases be provided by a software product's host operating system, should be implemented according to guidance in relevant standards such as the NIST SP800-90 series.

DEV.2.M290: Employ an approved entropy source

This mitigation is required to counter prediction of randomly generated values due to a weak entropy source

At Foundation Grade the product **is required to** use a raw entropy source whose entropy generation capability is understood.

The developer must provide a detailed description of the entropy source used, giving evidence that it can generate sufficient raw entropy for use in the device, including an estimate of entropy per bit. The entropy source should be implemented according to guidance in relevant standards such as the NIST SP800-90 series.

If a hardware noise source is used, then the manufacturer's name, the part numbers and details of how this source is integrated into the product must be supplied. If a software entropy source is employed, the API calls used must be provided. Where appropriate, details must be given of how the outputs of multiple entropy sources are combined.

DEV.2.M956: Encrypt communications traffic

This mitigation is required to counter interception of data from unencrypted communications

At Foundation Grade the product **is required to** use the MIKEY SAKKE Streaming Media Profile.

DEV.3 - Development >> Key Management

DEV.3.M134: State raw entropy requirements

This mitigation is required to counter prediction of randomly generated values due to a weak entropy source

At Foundation Grade the product **is required to** have clearly defined entropy requirements for all operational random number generation.

The developer must state how much raw entropy is required from product's entropy source (i.e. used to reseed the PRNG), based on analysis of all random numbers used in the product, including any generated keys.

At this grade, the amount of raw entropy required is likely to be at least:

- 256 bits for elliptic curve-based asymmetric mechanisms
- 128 bits for symmetric cryptographic mechanisms.

DEV.3.M140: Smooth output of entropy source with approved PRNG

This mitigation is required to counter prediction of randomly generated values due to a weak entropy source

This mitigation is required to counter prediction of randomly generated values due to insufficient raw entropy reaching the PRNG

At Foundation Grade the product **is required to** ensure that all random data used originates from a PRNG that is seeded by an approved entropy source.

The PRNG must also be reseeded sufficiently frequently to avoid raw entropy input being overused.

The PRNG and its reseeding mechanism, which could in certain cases be provided by a software product's host operating system, should be implemented according to guidance in relevant standards such as the NIST SP800-90 series.

DEV.3.M290: Employ an approved entropy source

This mitigation is required to counter prediction of randomly generated values due to a weak entropy source

At Foundation Grade the product **is required to** use a raw entropy source whose entropy generation capability is understood.

The developer must provide a detailed description of the entropy source used, giving evidence that it can generate sufficient raw entropy for use in the device, including an estimate of entropy per bit. The entropy source should be implemented according to guidance in relevant standards such as the NIST SP800-90 series.

If a hardware noise source is used, then the manufacturer's name, the part numbers and details of how this source is integrated into the product must be supplied. If a software entropy source is employed, the API calls used must be provided. Where appropriate, details must be given of how the outputs of multiple entropy sources are combined.

DEV.3.M808: Shared secret generation

This mitigation is required to counter capturing of key exchange messages to deduce shared secrets

At Foundation Grade the product **is required to** use the MIKEY SAKKE Streaming Media Profile.

DEV.3.M812: No Private Key export

This mitigation is required to counter export of private key material from the device

At Foundation Grade the product **is required to** use operating system mechanisms, such as user privileges and the operating system certificate store, or another protected certificate store, to ensure that unencrypted private keys or machine certificates cannot be retrieved by a standard user.

This must include protecting any APIs or interfaces added by the product which have access to the certificate store.

DEV.3.M816: Key distribution credentials revocation

This mitigation is required to counter using compromised key distribution credentials

At Foundation Grade the product **is required to** ensure that a compromised user or device can be prevented from rekeying.

DEV.3.M821: Key Exchange signature checks

This mitigation is required to counter spoofing key exchange messages

At Foundation Grade the product **is required to** verify signatures on key exchange messages and reject the request if the signature is invalid or the signing key is untrusted.

DEV.4 - Development >> Management

DEV.4.M267: Provide an automated configuration tool to enforce required settings

This mitigation is required to counter exploitation of an accidental misconfiguration

At Foundation Grade the product **is required to** be provided with a configuration tool, or other method, for an administrator to initially set it up into a suitable configuration.

If the product requires more than 12 options to be changed or set by an administrator to comply with these Security Characteristics, the developer must supply a tool or policy template which helps the administrator to achieve this in fewer steps.

DEV.4.M353: Ensure product security configuration can only be altered by an authenticated system administrator

This mitigation is required to counter unauthorised alteration of product's configuration

At Foundation Grade the product **is required to** ensure that only authenticated administrators are able to change the product's security enforcing settings.

3.2 Verification mitigations

VER.M80: Protocol robustness testing

This mitigation is required to counter discovery of a vulnerability in the implementation of the protocol stack

At Foundation Grade the evaluator **will** perform testing using commercial fuzzing tools.

Fuzz testing is described in more detail in The Process for Performing CPA Foundation Grade Evaluations.

VER.M347: Verify update mechanism

This mitigation is required to counter installing compromised software using the update process

At Foundation Grade the evaluator **will** validate the developer's assertions regarding the suitability and security of their update process.

The update process must provide a mechanism by which updates can be authenticated before they are applied.

The process and any configuration required must be documented within the Security Procedures.

VER.1 - Verify >> Encryption

VER.1.M4: Evaluation/Cryptocheck

This mitigation is required to counter exploitation of a cryptographic algorithm implementation error

At Foundation Grade the evaluator **will** ensure all cryptographic algorithms employed for security functionality have been validated as per the "Cryptography Review" section in The Process for Performing CPA Foundation Grade Evaluations.

This requirement also covers any PRNG implementation used by the product, if applicable.

VER.2 - Verify >> Key Management

VER.2.M4: Evaluation/Cryptocheck

This mitigation is required to counter exploitation of a cryptographic algorithm implementation error

At Foundation Grade the evaluator **will** ensure all cryptographic algorithms employed for security functionality have been validated as per the "Cryptography Review" section in The Process for Performing CPA Foundation Grade Evaluations.

This requirement also covers any PRNG implementation used by the product, if applicable.

3.3 Deployment mitigations

DEP.M39: Audit log review

This mitigation is required to counter exploitation of a software implementation/logic error

At Foundation Grade the deployment **is required to** regularly review audit logs for unexpected entries.

DEP.M131: Operating system verifies signatures

This mitigation is required to counter installation of a malicious privileged local service

At Foundation Grade the deployment **is required to** enable signature verification by the operating system for applications, services and drivers, where supported and where the product makes use of it.

DEP.M159: Update product

This mitigation is required to counter exploitation of a software implementation/logic error

At Foundation Grade the deployment **is required to** update to the latest version where possible.

DEP.M340: Address Space Layout Randomisation

This mitigation is required to counter exploitation of a software implementation/logic error

At Foundation Grade the deployment **is required to** enable ASLR in the host Operating System where available.

DEP.M348: Administrator authorised updates

This mitigation is required to counter installing compromised software using the update process

At Foundation Grade the deployment **is required to** confirm the source of updates before they are applied to the system.

The administrator is required to have authorised the updates before use. If an automatic process is used, the administrator must also configure the product to authenticate updates.

The update procedure to be used by the administrator must be described within the product's security procedures.

DEP.M813: Platform data encryption

This mitigation is required to counter extraction of sensitive data held on the device

At Foundation Grade the deployment **is required to** use the client on a host platform which provides user credential-backed encrypted storage accredited and configured for use with data classified at the same level as the maximum classification of communications data that will be used on the device.

DEP.M818: Host lockdown

This mitigation is required to counter exploitation of weakly configured host platform security settings

At Foundation Grade the deployment **is required to** configure the host platform according to CESG End User Devices Security Guidance for that platform.

DEP.M819: User authentication

This mitigation is required to counter exploitation of weakly configured host platform security settings

At Foundation Grade the deployment **is required to** deploy the client only on host operating systems which authenticate users before the device can be used.

DEP.1 - Deployment >> Session Handling

DEP.1.M809: Mixed classification state prevention

This mitigation is required to counter eavesdropping on a secure communications channel made while an insecure communications channel is active

At Foundation Grade the deployment **is required to** prohibit concurrent secure and non-secure communications with any applications on the platform.

DEP.1.M955: No additional secure real-time communications clients

This mitigation is required to counter exploitation of vulnerabilities caused by co-existence with other secure real-time communications clients

At Foundation Grade the deployment **is required to** ensure that any other secure real-time communications clients are disabled on the host.

DEP.2 - Deployment >> Key Management

DEP.2.M817: Secure private key distribution

This mitigation is required to counter interception of private keys during distribution

At Foundation Grade the deployment **is required to** use a private key distribution method which provides confidentiality, perfect forward secrecy and mutual authentication.

DEP.3 - Deployment >> Management

DEP.3.M38: Use automated configuration tool

This mitigation is required to counter exploitation of an accidental misconfiguration

At Foundation Grade the deployment **is required to** be configured using automated tools if provided.

DEP.3.M815: Compromise recovery

This mitigation is required to counter exploitation of compromised client before recovery processes enacted

At Foundation Grade the deployment **is required to** implement plans to stop a compromised device from rekeying on key expiry.

DEP.3.M820: Private key material expiry

This mitigation is required to counter using credentials of a user that have been unknowingly compromised

At Foundation Grade the deployment **is required to** perform regular rekeying of private key material as required by the MIKEY SAKKE Streaming Media Profile.

Appendix A Summary of changes to mitigations

CESG has changed the previous version of this document, named Secure VoIP Client Security Characteristic (version 2.0), with this version, named Secure real-time communications client (version 2.1), for the following reasons:

- Generalise the Security Characteristic to cover a multitude of different real time messaging applications instead of focusing on Voice over IP.
- Update the Security Characteristic to use the latest version of the CESG SC Libraries.

This has resulted in the following changes to mitigations.

A.1 Removed mitigations

The following mitigations have been removed.

- DEV.2.M141: Reseed PRNG as required (incorporated into DEV.2.M140)
- DEV.3.M141: Reseed PRNG as required (incorporated into DEV.3.M140)

A.2 Modified mitigations

The following mitigations have been modified.

- DEV.M349: Sanitise temporary variables (title also changed to: "DEV.M351: Erase sensitive data after use")
- DEV.M822: Secure host configuration
- DEV.1.M809: Mixed classification state prevention
- DEV.1.M811: Call security display (title also changed to: "DEV.1.M957: Communications security display")
- DEV.2.M138: State the Security Strength required for random numbers (title also changed to: "DEV.2.M134: State raw entropy requirements")
- DEV.2.M140: Smooth output of entropy source with approved PRNG
- DEV.2.M290: Employ an approved entropy source
- DEV.2.M810: Encrypt call traffic (title also changed to: "DEV.2.M956: Encrypt communications traffic")
- DEV.3.M138: State the Security Strength required for random numbers (title also changed to: "DEV.3.M134: State raw entropy requirements")
- DEV.3.M140: Smooth output of entropy source with approved PRNG
- DEV.3.M290: Employ an approved entropy source
- DEV.3.M808: Shared secret generation
- DEV.3.M821: Key Exchange signature checks
- VER.1.M4: Evaluation/Cryptocheck
- VER.2.M4: Evaluation/Cryptocheck
- DEP.M813: Platform data encryption
- DEP.M818: Host lockdown
- DEP.1.M809: Mixed classification state prevention
- DEP.1.M824: No additional VoIP clients numbers (title also changed to: " DEP.1.M955: No additional secure real-time communications clients")
- DEP.3.M820: Private key material expiry

A.3 Renamed mitigations

No mitigations have been renamed (except those that have been modified as well – see above).

A.4 New mitigations

No new mitigations have been added.

Appendix B References

This document references the following resources.

Label	Title	Location	Notes
[a]	The Process for Performing Foundation Grade CPA Evaluations	www.cesg.gov.uk/servicecatalogue/CPA	
[b]	RFC 3711 The Secure Real-time Transport Protocol	www.ietf.org/rfc/rfc3711.txt	
[c]	RFC 6509 MIKEY-SAKKE: Sakai-Kasahara Key Encryption in Multimedia Internet KEYing (MIKEY)	www.ietf.org/rfc/rfc6509.txt	
[d]	Technical Specification No. 63 – A MIKEY-SAKKE/SRTP Profile	www.cesg.gov.uk/publications	
[e]	End User Devices Security Guidance: <*platform*>	www.gov.uk	

Appendix C Glossary

The following definitions are used in this document.

Term	Definition
CPA	Commercial Product Assurance. A scheme run by CESG providing certificate-based assurance of commercial security products.
GAP	Government Assurance Pack. A framework for securing Windows 7 under an Active Directory Domain.
GPG	Good Practice Guide. A series of documents produced by CESG that provide guidance on specific aspects of IA.
IP	Internet Protocol
NPM	Non-Protectively Marked.
PRNG	Pseudo-random Number Generator
SAKKE	Saka-Kasahara Key Encryption
SC Map	Diagrammatic representation of a Security Characteristic (or part of one).
Security Characteristic	A standard which describes necessary mitigations which must be present in a completed product, its evaluation or usage, particular to a type of security product.
SIP	Session Initiation Protocol
SRTP	Secure Real-time Transport protocol
VoIP	Voice over IP