COMPARISON DOCUMENT BETWEEN THE SECURE AUDITING FOR LINUX SOFTWARE REQUIREMENTS AND THE COMMON CRITERIA

SAL

SECURE AUDITING FOR LINUX

August 2002

Table of Contents

1. Comparison to Common Criteria

1.1 *Background*

The Common Criteria (CC) was developed by a group of seven governmental associations as a criterion for evaluation of Information Technology (IT) security. This has become a standard for software security in countries such as Canada, France, Netherlands, Germany, United Kingdom and the United States. The National Security Agency and the National Institute of Standards and Technology are the two United States governmental organizations who are members of the "Common Criteria Project Sponsoring Organizations."

1.2 *Purpose*

This document compares our current development of the Secure Auditing for Linux Software Requirements Specifications (SAL – SRS) to the Common Criteria in order to meet government standards for secure auditing software.

The main purpose of this document is to target requirements of the SAL SRS, which need to be appended and/or adjusted with respect to requirements stated in the CC. To get more information on the Common Criteria, an online three-part document can be found at www.commoncriteria.org/cc/cc.html.

2. Table of SAL Evaluation

The following table is a listing of the Common Criteria requirements and mapping to the SAL-SRS document.

According to Common Criteria (Part II)	SAL Software Requirement Specification
Security Audit Automatic Response	
FAU_ARP.1.1	**REQUIREMENT NOT MET**
Security Audit Data Generation	
FAU_GEN.1.1	GEN010 + Table 2.1.2 of SRS
FAU_GEN.1.2	REP050 & REP080 & REP090
FAU_GEN.2.1	SAU020 / REP100
Security Audit Analysis	
FAU_SAA.1.1	REC010 – REC 200 (ALL) / Does Not Meet the "Potential Violation Part" of this standard.
FAU_SAA.1.2	**REQUIREMENT NOT MET**
FAU_SAA.2.1	REC170 – But No User Profile Storage
FAU_SAA.2.2	**REQUIREMENT NOT MET**
FAU_SAA.2.3	**REQUIREMENT NOT MET**
FAU_SAA.3.1	ENC040
FAU_SAA.3.2	**REQUIREMENT NOT MET**

FAU_SAA.3.3	**REQUIREMENT NOT MET**
FAU_SAA.4.1	**REQUIREMENT NOT MET**
FAU_SAA.4.2	**REQUIREMENT NOT MET**
FAU_SAA.4.3	**REQUIREMENT NOT MET**
Security Audit Review	
FAU_SAR.1.1	SAU020 / SAU030
FAU_SAR.1.2	REP050
FAU_SAR.2.1	GEN030
FAU_SAR.3.1	SAU030
Security Audit Event Selection	
FAU_SEL.1.1	SAU020
Security Audit Event Storage	
FAU_STG.1.1	GEN020
FAU_STG.1.2	GEN020
FAU_STG.2.1	GEN020
FAU_STG.2.2	GEN020
FAU_STG.2.3	PRF040 / PRF020
FAU_STG.3.1	CMP050 – To A Certain Extent
FAU_STG.4.1	**REQUIREMENT NOT MET**

3. Security Audit Requirements

The Security Audit Requirements in the CC can be found in the Class Family Audit (Class FAU). There are six components in the Class FAU. Of these six CC classes, three have been fully satisfied in our SAL-SRS document. The other three classes are partially or fully incomplete.

List below are the six classes, with the **bold** lettering representing incomplete families:

- **Security Audit Automatic Response (FAU_ARP)**
- Security Audit Data Generation (FAU_GEN)
- **Security Audit Analysis (FAU_SAA)**
- Security Audit Review (FAU_SAR)
- Security Audit Event Selection (FAU_SEL)
- **Security Audit Event Storage (FAU_STG)**

This document continues by discussing known class FAU areas that will be addressed in future revisions of SAL

3.1 *Security Audit Automatic Response*

FAU_ARP.1.1 **The TSF shall take [assignment: *list of the least disruptive actions*] upon detection of a potential security violation.**

We need to compile a list of actions that may pose a potential security violation. This will be addressed in future versions of our software.

3.2 *Security Audit Analysis*

FAU_SAA.1 Potential violation analysis

FAU_SAA.1.1 The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TSP.

This partially complies with REC010-REC200 from the SAL –SRS document. The inconsistent part is "potential violation." This will be addressed in future versions of our software.

FAU_SAA.1.2 The TSF shall enforce the following rules for monitoring audited events:

a) Accumulation or combination of [assignment: *subset of defined auditable events*] known to indicate a potential security violation;

b) [assignment: *any other rules*].

This action will comply once FAU_ARP 1.1 is satisfied. This will be addressed in future versions of our software.

FAU_SAA.2 Profile based anomaly detection

FAU_SAA.2.1 The TSF shall be able to maintain profiles of system usage, where an individual profile represents the historical patterns of usage performed by the member(s) of [assignment: *the profile target group*].

This action calls for the creation of individual profiles that are able to store a history of events. This will be addressed in future versions of our software.

FAU_SAA.2.2 The TSF shall be able to maintain a suspicion rating associated with each user whose activity is recorded in a profile, where the suspicion rating represents the degree to which the user's current activity is found inconsistent with the established patterns of usage represented in the profile.

A suspicion rating needs to be associated with each individual. This will be addressed in future versions of our software.

FAU_SAA.2.3 The TSF shall be able to indicate an imminent violation of the TSP when a user's suspicion rating exceeds the following threshold conditions [assignment: *conditions under which anomalous activity is reported by the TSF*].

This will be addressed in future versions of our software.

FAU_SAA.3 Simple attack heuristics

FAU_SAA.3.2 The TSF shall be able to compare the signature events against the record of system activity discernible from an examination of [assignment: *the information to be used to determine system activity*].

This will be addressed in future versions of our software.

FAU_SAA.3.3 The TSF shall be able to indicate an imminent violation of the TSP when a system event is found to match a signature event that indicates a potential violation of the TSP.

This will be addressed in future versions of our software.

FAU_SAA.4 Complex attack heuristics

FAU_SAA.4.1 The TSF shall be able to maintain an internal representation of the following event sequences of known intrusion scenarios [assignment: *list of sequences of system events whose occurrence are representative of known penetration scenarios*] and the following signature events [assignment: *a subset of system events*] that may indicate a potential violation of the TSP.

This will be addressed in future versions of our software.

FAU_SAA.4.2 The TSF shall be able to compare the signature events and event sequences against the record of system activity discernible from an examination of [assignment: *the information to be used to determine system activity*].

This will be addressed in future versions of our software.

FAU_SAA.4.3 The TSF shall be able to indicate an imminent violation of the TSP when system activity is found to match a signature event or event sequence that indicates a potential violation of the TSP.

This will be addressed in future versions of our software.

3.3 *Security Audit Event Storage*

FAU_STG.4.1 The TSF shall [selection: '*ignore auditable events', 'prevent auditable events, except those taken by the authorized user with special rights', 'overwrite the oldest stored audit records'*] and [assignment: *other actions to be taken in case of audit storage failure*] if the audit trail is full.

A possible solution is to set a threshold value. Additional solutions are still pending.