

Mobile Code Isolation Protection Profile

Issue : 0.5

Issue date : 9 May 2001

Status : Draft

Project/document reference : 363.20754.84.1

Authors : Steve H Hill
Kevin Appleford

Reviewed by : Andy Webber

Distribution : CESG
Logica

Table Of Contents

1	Introduction	3
1.1	Purpose	3
1.2	PP Overview	3
1.3	Related PPs.....	4
1.4	CC Conformance Claim	4
1.5	Glossary of Terms.....	4
1.6	Document Structure	5
1.7	References	6
2	TOE Description	7
2.1	Intended Use	7
2.2	Security Features.....	9
3	TOE Security Environment.....	11
3.1	Environmental and Method of Use Assumptions	11
3.2	Assumed Threats.....	12
3.3	Organisational security policies.....	13
4	Security Objectives	14
4.1	TOE Security Objectives.....	14
4.2	Security Objectives for the Environment	15
5	IT Security Requirements.....	16
5.1	TOE Security Functional Requirements	16
5.2	TOE Security Assurance Requirements.....	21
5.3	SOF claim.....	21
5.4	Security requirements for the IT environment	21
6	Protection Profile Rationale.....	22
6.1	Security Objectives Rationale.....	22
6.2	Security Requirements Rationale.....	26

1 Introduction

1.1 PP Identification

Title:	Mobile Code Isolation Protection Profile
Issue:	0.5
Publication date:	9 May 2001
CC Version:	2.1 (also known as ISO standard 15408)
Sponsoring organisation:	CESG

Other PP identification information required by [CC] can be found on the front page of this document. Related PPs are referenced in section 1.3.

1.2 PP Overview

This Protection Profile (PP) has been developed to identify and describe the security requirements needed for providing security when downloaded mobile code is executed on a user's desktop PC. The Target of Evaluation (TOE) for this PP is a desktop application which addresses this security need by isolating downloaded mobile code when it is run to prevent or limit damage.

For the purposes of this PP **mobile code** is defined as code that is downloaded from a remote source over a network and run on a user's PC with little or no intervention from the user. The most common types of mobile code are:

- scripting languages such as JavaScript and VB Script;
- Java;
- ActiveX controls;
- Macros within office automation applications such as word processors, spreadsheets and databases.

This PP defines requirements to monitor (and audit) accesses by mobile code to system resources, e.g. file system resources, network connections, processes, and so on. The TOE blocks the attempted access if it is considered to be potentially malicious. This determination is made on the basis of a configurable access control policy enforced by the TOE.

The security assurance requirements are specified as either EAL3 or EAL4. In this respect this (physical) PP document can be considered to contain two (logical) PPs that differ only in respect of the security assurance requirements.

Although this PP specifies requirements which make compliant and evaluated TOEs suitable for HMG installations, it does not require government only technologies, and may therefore be used in non HMG environments.

1.3 Related PPs

The Mobile Code Quarantine PP [QuarPP] and the Mobile Code Authentication PP [AuthPP] both also define security requirements to address the threat of malicious mobile code, and are closely related to this PP.

In particular, a desktop application which is claimed to be conformant with this PP may also be claimed conformant with the Mobile Code Authentication PP, since the operations it permits the mobile code to perform may depend, in part, on the degree of trust in that code as established through a check on its authenticity.

1.4 CC Conformance Claim

This PP is Part 2 conformant and Part 3 conformant for EAL3 or EAL4, as appropriate.

An ST claiming conformance with this PP shall clearly indicate at which level conformance is being claimed, e.g. by including a statement such as the following:

This ST claims conformance with the Mobile Code Isolation Protection Profile at the EAL4 level of assurance.

1.5 Glossary of Terms

This section contains definitions of technical terms that are used with a meaning specific to this document. Terms defined in the [CC] are not reiterated here, unless stated otherwise.

Denial of service attacks: These attacks aim to disrupt the normal operation of computer hardware or software to prevent its use. This commonly involves using so much of a particular resource that the response time becomes so slow as to make the resource effectively unusable, or that a critical component crashes.

Firewall: A firewall is a set of related programs and servers acting as a network gateway that protects the resources of a private network from users on other networks.

Gateway: A gateway is a network node that acts as an gateway to another network.

Mobile Code: Code that is downloaded and run with little or no user intervention. This definition excludes code manually downloaded from remote sites and run, e.g. via ftp. This is because, with manually downloaded code, the onus is on the user to ensure that the code is safe before running it, or that it runs in such a way as to minimise any damage it may cause.

Proxy: a proxy is an application that acts as an intermediary between a workstation user application and the Internet to provide security.

Trusted networks: Networks whose users are trusted not to:

- maliciously interfere with data travelling over the network,
- launch attacks on PCs on the network or
- passively offer malicious data to other PCs on the network.

Trusted networks are typically connected to one or more, larger networks, e.g. the Internet, in which there is less or no trust.

Untrusted networks: networks that are not trusted.

1.6 Document Structure

The structure of this document is as defined by [CC] Part 1 Annex C.

- Section 1 (this section) is the PP Introduction
- Section 2 is the TOE Description.
- Section 3 provides the statement of TOE security environment.
- Section 4 provides the statement of security objectives.
- Section 5 provides the statement of IT security requirements.

- Section 6 provides the PP rationale.

1.7 References

AuthPP	Mobile Code Authentication Protection Profile, 363.20754.83.1, Issue 0.6, April 2001
CC	Common Criteria for Information Technology Security Evaluation (Comprising Parts 1-3, [CC1], [CC2], [CC3]).
CC1	Common Criteria for Information Technology Security Evaluation Part 1: Introduction and General Model CCIMB-99-031, Version 2.1, August 1999.
CC2	Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements CCIMB-99-032, Version 2.1, August 1999.
CC3	Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements CCIMB-99-033, Version 2.1, August 1999.
IS1	HMG Infosec Standard No. 1 (IS1), April 1998
Memo21	CESG Infosec Memorandum No 21 Risk Management of Mobile Code, Issue 1.0, January 2001
QuarPP	Mobile Code Quarantine Protection Profile, 363.20754.85.1, Issue 0.5, May 2001

2 TOE Description

The Target of Evaluation (TOE) for this protection profile is a desktop application which isolates mobile code when it is run to prevent or limit damage. (This is often referred to as the 'sandbox' technique.)

2.1 Intended Use

There are a number of different approaches to securing end users against malicious mobile code, and it is likely that individuals, organisations and applications will implement a combination of these.

For organisations, the boundary between a trusted and untrusted network is typically controlled with a firewall or other filtering device. The aim is to prevent attackers on the untrusted network gaining access to information or resources on the trusted network by gaining control of a trusted PC or snooping on trusted network traffic. Figure 1 illustrates a typical network infrastructure. Firewalls or filtering devices may also be required to create boundaries between divisions within an organisation.

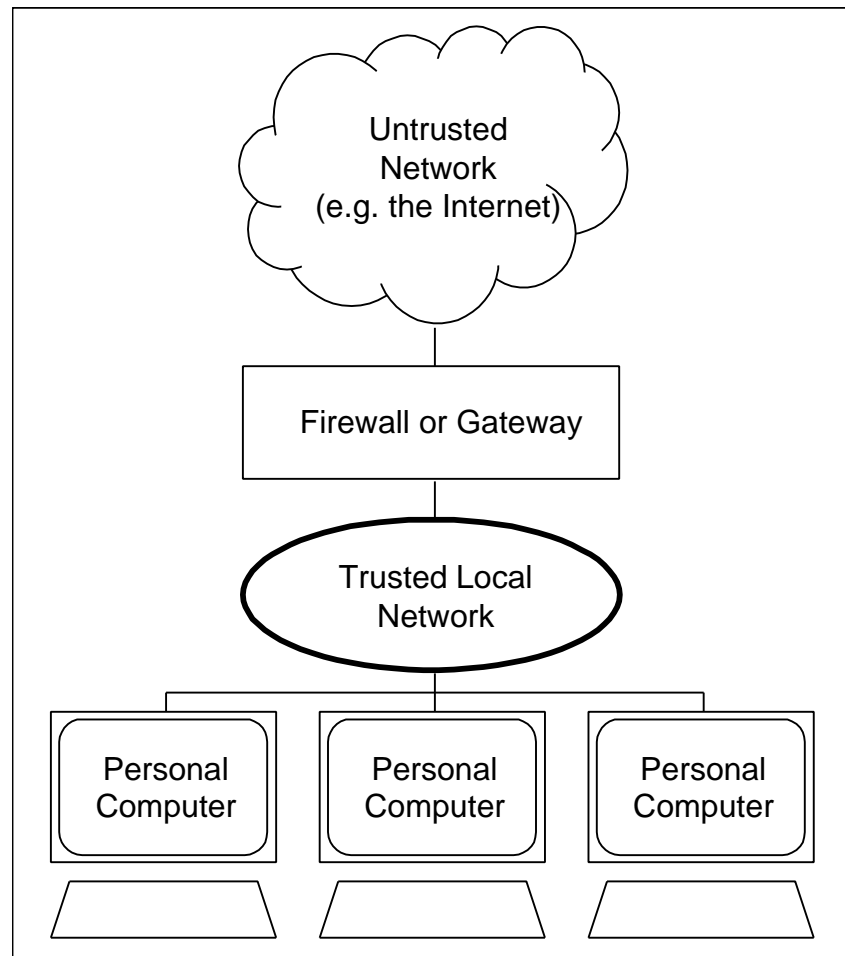


Figure 1: Typical Network Infrastructure

A correctly implemented and configured firewall will prevent most active attacks originating from the untrusted network. However, the assumed environment for the TOE described by this PP is one in which there is a business need to allow some mobile code through. This presents an opportunity to the attacker who may subvert the mobile code, or lure the unsuspecting user into executing malicious mobile code, so that it performs an attack when run on a workstation on the trusted network. The assets on the trusted network may also be placed at risk as a result of the execution of mobile code which has design or implementation flaws.

The TOE is to be installed on the PCs used by the users on the trusted network, and centrally managed by an administrator.

The boundary between trusted and untrusted networks is context dependent. For many organisations, it will be the connection between the organisation's network and the Internet. For some organisations boundaries exist between organisational divisions, e.g. to control access between research and development department and personnel departments. In some situations there is no local network, e.g. a user dialling into the Internet via an ISP where the PC connects directly to the untrusted network.

The TOE is not intended to provide protection against code manually downloaded from remote sites and run, e.g. via ftp. Rather, the onus is on a user manually downloading code to ensure it is safe before running it or to run it in such a way as to minimise potential damage it may cause.

Similarly, the TOE does not offer protection against denial of service attacks which aim to disrupt the normal operation of computer hardware or software to prevent its use, other than those which can be prevented by controlling access to system resources.

2.2 Security Features

The sole purpose of the TOE is to provide security: it includes no general IT features which are not in some way intended to counter the threat posed by mobile code executing on a user's PC.

The TOE allows mobile code to run on a user's PC in such a way as to prevent any malicious actions. Interpreted mobile code is run by the interpreter, which can identify malicious actions and refuse to execute them. Executable mobile code is constrained by wrapping all operating system calls with code that checks what operation is being requested and decides if it is potentially malicious.

This approach has significant advantages over the more traditional signature matching approaches since it affords protection against new instances of malicious mobile code, and does not rely on frequent database updates to maintain its effectiveness.

The TOE provides protection against the following types of potentially malicious mobile code:

- code executed by web browsers or other Internet applications, e.g. JavaScript, VBScript, ActiveX;
- macros within office automation applications such as word processors, spreadsheets and databases;
- executable files originating from the untrusted network.

The effectiveness of the TOE's security features is thus dependent on its configuration, such that potentially malicious access to system resources (e.g. file system resources, network connections, processes, and so on) is blocked. Key configuration decisions are left in the hands of an authorised central administrator. However, some decisions may be delegated to the end-user.

To assist the administrator in the fine-tuning of the access control rules, the TOE also provides the capability to audit accesses by downloaded mobile code to system resources.

3 TOE Security Environment

The statement of TOE security environment describes the security aspects of the environment in which the TOE is intended to be used, and the manner in which it is expected to be employed.

To this end, the statement of TOE security environment identifies and lists the assumptions made on the environment and the intended method of use of the TOE, defines the threats that the TOE is designed to counter, and the organisational security policies with which the TOE is designed to comply.

3.1 Environmental and Method of Use Assumptions

3.1.1 Method of use assumptions

A.ALTSOURCE It is assumed that the environment provides appropriate protection against damage to assets by code that might be introduced to the trusted network by means other than being received directly from the untrusted network.

This assumption precludes threats posed to the assets by malicious code that is not delivered directly from the untrusted network, e.g. imported from a floppy disk or CD-ROM. If, however, such threats are addressed by the TOE then this assumption is not relevant, and the statement of threats should be modified accordingly in the ST.

A.TRUSTCODE It is assumed that code originating from the trusted network is not malicious.

This assumption scopes the statement of threats to that posed by mobile code originating from the untrusted network. If the conformant TOE offers protection against mobile code regardless of its origin then this assumption is not relevant.

3.1.2 Personnel assumptions

A.TRUSTUSER It is assumed that users on the trusted network are trustworthy individuals.

The intent of this assumption is to assert that threats posed by malicious users who might intentionally introduce malicious mobile code into the trusted network are outside the scope of the security problem to be addressed. Thus the threats are limited to the unintentional downloading of malicious mobile code by users due to negligence, naivety or ignorance.

A.TRUSTADMIN It is assumed that administrators are competent and trustworthy individuals.

This assumption precludes threats that might otherwise be posed by negligent or hostile administrators of the TOE.

3.2 Assumed Threats

This section describes the threats to the assets that require protection.

3.2.1 Assets

The primary assets of concern to this PP are the information and resources on the trusted network, the integrity and availability of which is important to the organisation which owns them. Specific data assets on the trusted network may also require protection of their confidentiality against more directed attacks, e.g. user passwords, secret or private cryptographic keys, personal information, and so on.

3.2.2 Threat agents

The *source* of the threat is malicious mobile code downloaded from the untrusted network and executing on a user's PC. The *threat agents* are attackers on the untrusted network who write malicious mobile code. They are assumed to have various levels of expertise, motivation and resources. Their motivation may arise from a variety of reasons, including a desire to:

- impede or interfere with the operation of specific organisations;
- gain financially through attack directed against individuals rather than organisations, e.g. to obtain credit card details;
- seek publicity or gain notoriety.

3.2.3 Threat descriptions

The general threat to be countered is that of malicious mobile code compromising integrity and availability of information and resources, as well as possibly the confidentiality of information (e.g. user passwords, cryptographic keys, or other personal information). For the purposes of the statement of threats in the PP, 'mobile code' excludes any code which requires explicit user intervention before it executes.

The specific threats to be countered are as follows. Note that all concern the **direct** delivery of mobile code from the untrusted network, e.g. through applications such as web browsers, email clients, and news readers. Indirect delivery of such code (e.g. from CD-ROM) is precluded by the assumption A.ALTSOURCE. If, however, the conformant TOE offers protection against such threats then the statement of TOE security environment should be revised accordingly in the ST.

T.BROWSER A compromise of assets occurs as a result of the execution of mobile code by Internet applications (e.g. web browsers) on a user's desktop PC, workstation or laptop.

This threat relates to the execution of malicious mobile code such as JavaScript, VBScript and ActiveX.

T.SUSPECTDOC A compromise of assets occurs as a result of the execution of malicious macros in documents, received directly from the untrusted network, on a user's desktop PC, workstation or laptop.

This threat includes the execution of malicious macros contained in documents produced by office automation applications such as word processors, spreadsheets and databases. The threat here primarily concerns those macros that execute automatically when a user opens the document.

T.SUSPECTEXE A compromise of assets occurs as a result of the execution of executable files, received directly from the untrusted network, on a user's desktop PC, workstation or laptop.

This threat covers the execution of any potentially suspect executable file that a user on the trusted network receives from the untrusted network, e.g. downloaded from a website or received as an email attachment.

3.3 Organisational security policies

The following organisational security policy (OSP) is to be met by the TOE.

P.EAL IT countermeasures shall be assured to a level equal to EAL3 as a minimum.

4 Security Objectives

4.1 TOE Security Objectives

O.ISOLATION The TOE shall be able to flag as 'suspect' all mobile code delivered directly from the untrusted network, and shall be able to constrain the activities of such code to a configurable set of operations that are considered to be non-malicious. This shall include binary executable files, documents containing macros which execute automatically, and Internet applications which execute mobile code.

This is the principal security objective of the TOE, which is to enforce an access control policy on the activity of 'suspect' mobile code. To enforce this policy, the TOE must be able to keep track of all such code received directly from the untrusted network. The decision as to whether or not a particular operation is malicious depends on the configuration of this policy.

O.CONFIGURE The TOE shall be able to constrain the ability to configure the mobile code access control policies to those individuals who are authorised to do so.

O.ISOLATION identifies the need for a *configurable* access control policy. This security objective ensures the TOE is able to control who can modify this policy. In general, the ability to define or modify the policy should be constrained to an authorised administrator. However, it is possible that some decisions can be safely delegated to the end-user.

O.AUDIT The TOE shall be able to record attempts by downloaded mobile code to perform operations that are identified as potentially malicious.

This security objective is intended to assist both retrospective analysis (in the event of a compromise of assets by downloaded mobile code) and also to support fine-tuning of the mobile code access control policy.

O.ENFORCE The TOE shall ensure that the downloaded mobile code cannot circumvent the access control checks.

This security objective identifies the need to ensure that the mobile code access control checks cannot be bypassed, and that the mobile code cannot interfere in any way with the operation of the TOE.

O.EAL The TOE security functions shall be assured to EAL3 as a minimum.

This security objective is to be addressed by TOE security assurance requirements rather than TOE security functional requirements, and is driven by the P.EAL OSP.

4.2 Security Objectives for the Environment

O.ALTSOURCE Those responsible for the TOE environment shall establish and implement procedures to address the threat posed by malicious code that was not received directly from the untrusted network.

This security objective is necessary since the TOE does not protect the assets against malicious code introduced indirectly from the untrusted network, e.g. by importing from floppy disk or CD-ROM. However, if the conformant TOE does offer such protection (e.g. by enabling a user to explicitly request that any specified executable file is run in a 'safe mode', regardless of origin) then this security objective is not relevant, and the statement of security objectives should be modified accordingly in the ST.

O.AUDITLOG Administrators of the TOE shall ensure that the audit log is analysed on a regular basis, and that it is archived at appropriate intervals.

This security objective helps ensure the continued effectiveness of the audit measures implemented by the TOE.

O.SECPOL Administrators shall ensure that the TOE is configured in such a way as to meet the specific security needs of the organisation and the environment in which it is deployed.

The specific configuration of the TOE's access control policy rules (see O.CONFIGURE) should reflect the security policies of the organisation in which the TOE is deployed. This means that a security policy must first be established to identify what mobile code is considered to be 'suspect', and to define what operations such code can be allowed to perform. This objective also identifies the need to continually keep the security policy under review, and to fine-tune the configuration accordingly.

O.TRUSTADMIN Those responsible for the TOE environment shall establish and implement procedures for establishing trust and competence in administrators of the TOE.

This objective may be achieved through the vetting and training of administrators.

O.USER_ED Those responsible for the TOE environment shall ensure that users on the trusted network are trustworthy and are made aware of the dangers of executing mobile code from an untrusted source.

This security objective identifies the need to ensure users are made aware of the dangers posed by mobile code, and the importance of following the organisation's security operating procedures. Procedural or personnel (e.g. vetting) measures are also needed to reduce the risk of users intentionally introducing malicious mobile code onto the trusted network.

5 IT Security Requirements

5.1 TOE Security Functional Requirements

This section identifies the security functional requirements (SFRs) required of the TOE to meet its security objectives.

The components taken from [CC2] to specify the SFRs are listed in the table below together with an indication of whether the components are *iterated* (indicated by “(*N)” where N identifies the number of iterations) or *refined*. (See [CC2, 2.1.4] for an explanation of permitted operations on functional components.)

Assignment and selection operations to be completed by the ST author are indicated using the same notation as used in [CC2]. Partially completed operations are denoted by the use of *italicised text* for the key word *assignment* or *selection* (as appropriate). Completed assignment and selection operations are indicated by *italicised text*. Refinements of components are indicated by **emboldened text**.

CLASS	FAMILY	COMPONENT	REFINED?
FAU	FAU_GEN	FAU_GEN.1	
	FAU_SAR	FAU_SAR.1	
	FAU_STG	FAU_STG.1	
FDP	FDP_ACC	FDP_ACC.1	
	FDP_ACF	FDP_ACF.1	Y
FIA	FIA_UAU	FIA_UAU.1	Y
	FIA_UID	FIA_UID.1	Y
FMT	FMT_MSA	FMT_MSA.1	
	FMT_SMR	FMT_SMR.1	

CLASS	FAMILY	COMPONENT	REFINED?
FPT	FPT_RVM.1	FPT_RVM.1	Y
	FPT_SEP	FPT_SEP.1	
	FPT_STM	FPT_STM.1	

Table 1 – TOE Security Functional Requirements

5.1.1 Security Audit (FAU)

5.1.1.1 Audit data generation (FAU_GEN.1)

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

- a) Start-up and shutdown of the audit functions;
- b) *All requests by downloaded mobile code to perform an operation on a system resource.*
- c) [assignment: *other specifically defined auditable events*].

Application note: As a minimum the events listed at a) and b) must be auditable. The latter is equivalent to the requirement for the 'basic' level of audit for FDP_ACF.1 as defined in [CC2]. However, this PP does not mandate the full set of audit events for the 'basic' level: the selection has thus been completed by the choice of the 'not specified' level.

Note that in this and other SFRs in this section, the term 'downloaded mobile code' is used as a convenient shorthand for 'mobile code delivered directly from the untrusted network'.

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

- a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
- b) For each audit event type, based on the auditable event definitions of the functional components included in the PP, [assignment: *other audit relevant information*].

Application note: The assignment at paragraph b) should be completed to specify any additional information recorded. An assignment of 'none' is acceptable.

5.1.1.2 Audit review (FAU_SAR.1)

FAU_SAR.1.1 The TSF shall provide *authorised administrators* with the capability to read *all audit information* from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

5.1.1.3 Protected audit trail storage (FAU_STG.1)

FAU_STG.1.1 The TSF shall protect the stored audit records from unauthorised deletion.

FAU_STG.1.2 The TSF shall be able to [selection: *prevent*, *detect*] modifications to the audit records.

Application note: This SFR means that the audit trail should be stored centrally rather than locally on the user's PC.

5.1.2 User Data Protection (FDP)

5.1.2.1 Subset access control (FDP_ACC.1)

FDP_ACC.1.1 The TSF shall enforce the *mobile code access control policy* on:

- a) *subjects: downloaded mobile code*
- b) *objects: system resources*
- c) *operations: attempted accesses to system resources by downloaded mobile code.*

Application note: The TOE summary specification for the conformant ST should enumerate the system resources protected by the TOE, e.g. file system resources, network connections, processes, and so on.

5.1.2.2 Security attribute based access control (FDP_ACF.1)

FDP_ACF.1.1 The TSF shall enforce the *mobile code access control policy* to **system resources** based on [assignment: *security attributes*, *named groups of security attributes*].

Application note: This element has been refined to clarify the type of objects covered by the access control rules. The assignment should be completed to clearly identify the attributes used as the basis for defining the mobile code access control rules, e.g. type of system resource, type of operation.

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among **downloaded mobile code** and **system resources** is allowed:

- *Mobile code is permitted to perform an operation only if it is determined by the TOE not to be a potentially malicious action.*

Application note: This element has been refined to clarify the subjects and objects covered by the access control rules.

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [*assignment: rules, based on security attributes, that explicitly authorise access of downloaded mobile code to system resources*].

Application note: The assignment should be completed to specify any exceptions to the mobile code access control policy rules. For example, the TOE may permit mobile code which has been authenticated or explicitly added to a “white list” to run outside the sandbox. An assignment of ‘none’ is acceptable.

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the [*assignment: rules, based on security attributes, that explicitly deny access of downloaded mobile code to system resources*].

Application note: The assignment should be completed to specify any exceptions to the mobile code access control policy rules. An assignment of ‘none’ is acceptable.

5.1.3 Identification and Authentication (FIA)

5.1.3.1 Administrator authentication (FIA_UAU.1)

FIA_UAU.1.1 The TSF shall allow [*assignment: list of TSF mediated actions*] on behalf of the **administrator** to be performed before the **administrator** is authenticated.

FIA_UAU.1.2 The TSF shall require each **administrator** to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that **administrator**.

Application note: This component has been refined to clarify that authentication is only required for assumption of the administrator role.

5.1.3.2 Administrator identification (FIA_UID.1)

FIA_UID.1.1 The TSF shall allow [*assignment: list of TSF mediated actions*] on behalf of the **administrator** to be performed before the **administrator** is identified.

FIA_UID.1.2 The TSF shall require each **administrator** to be successfully identified before allowing any other TSF-mediated actions on behalf of that **administrator**.

Application note: This component has been refined to clarify that identification is only required for assumption of the administrator role.

5.1.4 Security Management (FMT)

5.1.4.1 Management of security attributes (FMT_MSA.1)

FMT_MSA.1.1 The TSF shall enforce the *mobile code access control policy* to restrict the ability to modify the *mobile code access control policy rules* to the *authorised administrator* and [*assignment: the authorised identified roles*].

Application note: The roles permitted to modify the mobile code access control policy rules must be defined in the ST. Normally, it would be expected that only authorised administrators are permitted to modify the policy rules. However, it is possible that some aspects may be safely delegated to normal users. It is for the ST to justify the policy implemented by the TOE, and to demonstrate conformance with the O.CONFIGURE security objective. If only the administrator can configure the access control rules then an assignment of 'none' should be made; it is recommended that, for clarity, this is made implicitly by deleting the words after 'authorised administrator'.

5.1.4.2 Security roles (FMT_SMR.1)

FMT_SMR.1.1 The TSF shall maintain the roles:

- a) *the authorised administrator*
- b) [assignment: other authorised identified roles].

Application note The assignment at paragraph b) should be completed in the same way as for FMT_MSA.1.

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

5.1.5 Protection of TOE Security Functions (FPT)

5.1.5.1 Non-bypassability of the TOE Security Policy (FPT_RVM.1)

FPT_RVM.1.1 The TSF shall ensure that **the mobile code access control** functions are invoked and succeed before **each attempted operation by downloaded mobile code** within the TSC is allowed to proceed.

Application note: The component has been refined to emphasise that the SFR is concerned with prevention of bypass of the mobile code isolation function.

5.1.5.2 TSF domain separation (FPT_SEP.1)

FPT_SEP.1.1 The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.1.2 The TSF shall enforce separation between the security domains of subjects in the TSC.

5.1.5.3 Reliable timestamps (FPT_STM.1)

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps for its own use.

5.2 TOE Security Assurance Requirements

As stated in Section 1.2, the security assurance requirements are specified as either EAL3 or EAL4, as defined in [CC3].

The effect of this approach is that this (physical) PP document contains two (logical) PPs that differ only in respect of the EAL mandated. As stated in Section 1.4, an ST claiming conformance to the Mobile Code Isolation Protection Profile shall clearly state the level at which conformance is claimed.

5.3 SOF claim

The minimum SOF rating required by this PP is **SOF-medium**.

5.4 Security requirements for the IT environment

No security requirements are specified for the IT environment.

6 Protection Profile Rationale

This section demonstrates the suitability of the choice of security objectives and security requirements to address the security needs posed by the statement of TOE security environment.

6.1 Security Objectives Rationale

This section demonstrates how the threats are countered, and the organisational security policies and assumptions are met, by the security objectives.

6.1.1 Security objectives are suitable to counter the threats

The table below shows that how each threat is countered by at least one security objective.

Threats	Objectives									
	O.ISOLATION	O.CONFIGURE	O.AUDIT	O.ENFORCE	O.EAL	O.ALTSOURCE	O.AUDITLOG	O.SECPOL	O.TRUSTADMIN	O.USER_ED
T.BROWSER	x	x	x	x	x		x	x	x	x
T.SUSPECTDOC	x	x	x	x	x		x	x	x	x
T.SUSPECTEXE	x	x	x	x	x		x	x	x	x

Table 2 Correlation between Threats and Objectives

T.BROWSER *A compromise of assets occurs as a result of the execution of mobile code by Internet applications (e.g. web browsers) on a user's desktop PC, workstation or laptop.*

O.ISOLATION ensures that mobile code executed by Internet applications, such as web browsers, is noted as being suspect, and that code so identified is constrained to a set of operations that are considered non-malicious. This prevents code such as JavaScript, VBScript and ActiveX from damaging the assets represented by the system resources.

O.ISOLATION is supported by the following security objectives:

- O.ENFORCE ensures that the access control checks cannot be bypassed or otherwise circumvented.
- O.CONFIGURE ensures that the ability to identify non-malicious operations is restricted to those individuals that are explicitly authorised and suitably trained in accordance with O.TRUSTADMIN and (to the extent that any responsibilities are delegated to end-users) O.USER_ED.
- O.SECPOL supports O.CONFIGURE by ensuring that a security policy is established that is appropriate to the security needs of the specific organisation and environment; the TOE is then configured in accordance with this policy.
- O.AUDIT supports O.CONFIGURE by helping the administrators to identify and fine tune the list of operations that may be safely executed. This is in turn supported by O.AUDITLOG which ensures that the audit trail is adequately maintained and reviewed.

O.EAL supports the other objectives of the TOE by ensuring that the security functions are suitably assured.

T.SUSPECTDOC *A compromise of assets occurs as a result of the execution of malicious macros in documents received directly from the untrusted network on a user's desktop PC, workstation or laptop.*

O.ISOLATION ensures that downloaded documents which may contain malicious macros are noted as being suspect, and that macros executed by the relevant office automation application are constrained to a set of operations that are considered to be non-malicious. This prevents macros from damaging the assets represented by the system resources.

O.ISOLATION is supported by O.ENFORCE, O.CONFIGURE, O.AUDIT, O.TRUSTADMIN, O.USER_ED and O.SECPOL as described in the rationale for T.BROWSER above.

O.EAL supports the other objectives of the TOE by ensuring that the security functions are suitably assured.

T.SUSPECTEXE *A compromise of assets occurs as a result of the execution of executable files received directly from the untrusted network on a user's desktop PC, workstation or laptop.*

O.ISOLATION ensures that downloaded executable files are noted as being suspect, and that code so identified is constrained to a set of operations that are considered non-malicious. This prevents such executable files from damaging the assets represented by the system resources.

O.ISOLATION is supported by O.ENFORCE, O.CONFIGURE, O.AUDIT, O.TRUSTADMIN, O.USER_ED and O.SECPOL as described in the rationale for T.BROWSER above.

O.EAL supports the other objectives of the TOE by ensuring that the security functions are suitably assured.

6.1.2 Security objectives are suitable to meet the OSPs

The table below shows that each organisational security policy is covered by at least one security objective.

OSP	Objectives									
	O.ISOLATION	O.CONFIGURE	O.AUDIT	O.ENFORCE	O.EAL	O.ALTSOURCE	O.AUDITLOG	O.SECPOL	O.TRUSTADMIN	O.USER_ED
P.EAL					x					

Table 3 Correlation between OSPs and Objectives

P.EAL *IT countermeasures shall be assured to a level equal to EAL3 as a minimum.*

O.EAL upholds P.EAL directly.

6.1.3 Security objectives are suitable to uphold the assumptions

The table below shows that each assumption is covered by security objectives for the environment.

Threats	Objectives									
	O.ISOLATION	O.CONFIGURE	O.AUDIT	O.ENFORCE	O.EAL	O.ALTSOURCE	O.AUDITLOG	O.SECPOL	O.TRUSTADMIN	O.USER_ED
A.ALTSOURCE						x				x
A.TRUSTCODE						x				x
A.TRUSTUSER										x
A.TRUSTADMIN									x	

Table 4 Correlation between Assumptions and Objectives

A.ALTSOURCE It is assumed that the environment provides appropriate protection against damage to assets by code that might be introduced to the trusted network by means other than being received directly from the untrusted network.

O.ALTSOURCE upholds A.ALTSOURCE directly. O.USER_ED also supports O.ALTSOURCE by ensuring that end-users have appropriate awareness of the dangers posed by malicious mobile code, regardless of where it originates from or its method of delivery to the trusted network.

A.TRUSTCODE *It is assumed that code originating from the trusted network is not malicious.*

A sub-objective arising out of the objective O.ALTSOURCE will be to ensure that those responsible for the TOE environment shall establish and implement procedures for establishing trust in the users of the TOE, otherwise malicious users could introduce malicious code.

O.USER_ED provides support by ensuring that users are aware of the dangers of executing mobile code from an untrusted source. The additional aspects of O.ALTSOURCE ensure that the threat posed by the introduction of other forms of malicious code to the trusted network is addressed by suitable procedures, and that they are implemented.

Thus, together these objectives uphold A.TRUSTCODE.

A.TRUSTUSER *It is assumed that users on the trusted network are trustworthy individuals.*

O.USER_ED ensures that end-users are aware of the dangers posed by malicious mobile code, and that procedures are in place to ensure that they do not intentionally introduce such code onto the trusted network.

A.TRUSTADMIN *It is assumed that administrators are competent and trustworthy individuals.*

O.TRUSTADMIN upholds A.TRUSTADMIN directly.

6.2 Security Requirements Rationale

This section demonstrates that the security requirements are appropriate, both in terms of their suitability to achieve the security objectives, and in collectively providing a mutually supportive and internally consistent whole.

6.2.1 Security requirements are suitable to achieve the security objectives

This section provides the correlation and justification of suitability between the objectives and the Security Functional Requirements.

TOE Security Objectives	Security Requirement
O.ISOLATION	Subset access control FDP_ACC.1 Security attribute based access control FDP_ACF.1
O.CONFIGURE	Administrator authentication FIA_UAU.1 Administrator identification FIA_UID.1 Management of security attributes FMT_MSA.1 Security roles FMT_SMR.1

TOE Security Objectives	Security Requirement
O.AUDIT	Audit data generation FAU_GEN.1 Audit review FAU_SAR.1 Protected audit trail storage FAU_STG.1 Reliable timestamps FPT_STM.1
O.ENFORCE	Non-bypassability of the TOE Security Policy FPT_RVM.1 TSF domain separation FPT_SEP.1
O.EAL	EAL3 or EAL4, as appropriate.

Table 5 - Correlation between Objectives for the TOE and SFRs

O.ISOLATION

The TOE shall be able to flag as 'suspect' all mobile code delivered directly from the untrusted network, and shall be able to constrain the activities of such code to a configurable set of operations that are considered to be non-malicious. This shall include binary executable files, documents containing macros which execute automatically, and Internet applications which execute mobile code.

FDP_ACC.1 ensures that the mobile code access control policy is enforced on accesses to system resources by downloaded mobile code, which thus achieves the objective of flagging such code as 'suspect'.

FDP_ACF.1 ensures that any operation on a system resources by mobile code marked as 'suspect' is forbidden if the TOE is able to determine that it is in the set of operations identified as potentially malicious. FDP_ACF.1.3 introduces the possibility of some mobile code being exempted from the access control checks; however, such code is then explicitly authorised to do so, and hence is no longer considered to be 'suspect'.

O.CONFIGURE

The TOE shall be able to constrain the ability to configure the mobile code access control policies to those individuals who are authorised to do so.

FMT_MSA.1 ensures that the mobile code access policy rules can only be modified by those individuals who are authorised to do so (normally the authorised administrator).

FMT_SMR.1 ensures that the role of administrator is defined and that users may be associated with such roles.

FIA_UAU.1 and FIA_UID.1 ensure that administrators of the TOE are identified and authenticated, thereby supporting FMT_MSA.1.

O.AUDIT

The TOE shall be able to record attempts by downloaded mobile code to perform operations that are identified as potentially malicious.

FAU_GEN.1 ensures that audit records can be generated for attempted operations on system resources by suspect mobile code, including those that have been identified as being malicious. FPT_STM.1 ensures that these audit records are reliably time-stamped.

FAU_SAR.1 ensures that the audit records may be read by administrators in a useable fashion.

FAU_STG.1 ensures that the integrity of the stored audit records is protected.

O.ENFORCE

The TOE shall ensure that the downloaded mobile code cannot circumvent the access control checks.

FPT_RVM.1 ensures that the mobile code access control policy which controls access to system resources by downloaded mobile code is always invoked and succeeds before mobile code is allowed to access system resources, i.e. the checks cannot be bypassed.

FPT_SEP.1 ensures that the TSF maintains a separate security domain for its execution that protects it from interference and tampering that might otherwise permit circumvention of the mobile code access control policy.

6.2.2 Security requirements are mutually supportive

6.2.2.1 Dependency analysis

The following table shows the dependencies between the SFRs. The rows list all the components included in this PP to specify the SFRs. The columns identify the dependencies of those components as specified in [CC2].

PP component	FDP_ACC.1	FDP_ACF.1	FIA_UID.1	FMT_MSA.3	FMT_SMR.1	FAU_GEN.1	FPT_STM.1
FDP_ACC.1		x					
FDP_ACF.1	x			x			
FIA_UAU.1			x				
FIA_UID.1							
FMT_MSA.1	x				x		
FMT_SMR.1			x				
FAU_GEN.1							x
FAU_SAR.1						x	
FAU_STG.1						x	
FPT_STM.1							
FPT_RVM.1							
FPT_SEP.1							

Table 6 Dependencies between SFRs

Key x – direct dependencies
 i – indirect dependencies
 o – optional dependencies

All dependencies highlighted in the above table are satisfied by the TOE SFRs, with the following exception:

- The dependency of FMT_MSA.3 does not need to be satisfied because there is no notion of the TOE supporting the creation of system resources, and hence assigning default security attribute values on creation.

Dependencies between security assurance requirements are satisfied because they are defined in terms of a self-contained assurance package (EAL3 or EAL4 as appropriate), with no augmentations.

6.2.2.2 Security requirements form a mutually supportive and consistent whole

The SFRs consist of a main group of mutually supportive access control requirements which in their turn are supported by security audit and security management requirements. This whole group is underpinned by requirements that ensure that they are always invoked, and protect their integrity. Thus:

1. FDP_ACC.1 and FDP_ACF.1 ensure that the principal security objective of the TOE is achieved, i.e. to constrain the operations that may be performed by downloaded mobile code unless they have been explicitly exempted.
2. FMT_MSA.1 and FMT_SMR.1 ensure that the role of administrator exists and that only administrators may modify the mobile code access control rules, in support of FDP_ACC.1 and FDP_ACF.1. FIA_UAU.1 and FIA_UID.1 ensure that administrators are identified and authenticated before performing their duties as such, thus supporting FMT_MSA.1 and FMT_SMR.1. These requirements act together to constrain the ability to configure the mobile code to those authorised so to do, supporting the main objective.
3. FAU_GEN.1, FAU_SAR.1, FAU_STG.1 and FPT.STM.1 support the configuration of the mobile access control rules by providing the creation and review of auditing events which will assist the administrator in reviewing and fine-tuning the security policy.
4. FPT_RVM.1 and FPT_SEP.1 support the other SFRs by ensuring that they cannot be bypassed or tampered with by untrusted subjects.

The security assurance requirements comprise a set of mutually supportive requirements as they correspond to a self-contained assurance package (EAL3 or EAL4 as appropriate). All such requirements by definition support the SFRs by providing confidence that the TOE satisfies those requirements.

Thus, the security requirements act together forming a supportive and consistent whole.

6.2.3 Security assurance requirements are appropriate

The minimum evaluation assurance level for this PP, namely EAL3, is selected because it trivially satisfies the security objective O.EAL (which is, itself, mandated by the OSP P.EAL).

An assurance level of EAL4 clearly satisfies this minimum requirement. It is identified as an option in this PP document since it may be more appropriate for particular types of environment. EAL4 represents the highest EAL that can be practically achieved without requiring substantial specialist knowledge, skills and other resources.

6.2.4 Strength of Function claim is appropriate

The SOF claim of **SOF-medium** is consistent with the security objectives since it is commensurate with an assurance level of EAL3 or EAL4. The appropriateness of the assurance level has already been justified.