PP-Module for Web Browsers

Version: 1.1
2023-08-25
National Information Assurance Partnership

Revision History

Version	Date	Comment
1.0	2021-06-18	Initial release as PP-Module
1.1	2023-08-25	Updates to conform to CC:2022

Contents

1 Introduction

1.1 Overview

The scope of the PP-Module for Web Browsers, Version 1.1 is to describe the security functionality of web browser applications in terms of [CC] and to define functional and assurance requirements for the product-specific capabilities of web browser applications. Web browsers are client applications that retrieve and render content provided by web servers, primarily using the hypertext transfer protocol (HTTP) or HTTP Secure (HTTPS). This PP-Module is intended for use with the following Base-PP:

- Protection Profile for Application Software, Version 2.0

This Base-PP is valid because web browsers are a specific type of software application.

1.2 Terms

The following sections list Common Criteria and technology terms used in this document.

1.2.1 Common Criteria Terms

Assurance	Grounds for confidence that a TOE meets the SFRs [CC].
Base Protection Profile (Base-PP)	Protection Profile used as a basis to build a PP-Configuration.
Collaborative Protection Profile (cPP)	A Protection Profile developed by international technical communities and approved by multiple schemes.
Common Criteria (CC)	Common Criteria for Information Technology Security Evaluation (International Standard ISO/IEC 15408).
Common Criteria Testing Laboratory	Within the context of the Common Criteria Evaluation and Validation Scheme (CCEVS), an IT security evaluation facility accredited by the National Voluntary Laboratory Accreditation Program (NVLAP) and approved by the NIAP Validation Body to conduct Common Criteria-based evaluations.
Common Evaluation Methodology (CEM)	Common Evaluation Methodology for Information Technology Security Evaluation.
Extended Package (EP)	A deprecated document form for collecting SFRs that implement a particular protocol, technology, or functionality. See Functional Packages.
Functional Package (FP)	A document that collects SFRs for a particular protocol, technology, or functionality.
Operational Environment (OE)	Hardware and software that are outside the TOE boundary that support the TOE functionality and security policy.
Protection Profile (PP)	An implementation-independent set of security requirements for a category of products.
Protection Profile Configuration (PP-Configuration)	A comprehensive set of security requirements for a product type that consists of at least one Base-PP and at least one PP-Module.
Protection Profile Module (PP-Module)	An implementation-independent statement of security needs for a TOE type complementary to one or more Base-PPs.
Security Assurance Requirement (SAR)	A requirement to assure the security of the TOE.
Security Functional Requirement (SFR)	A requirement for security enforcement by the TOE.
Security	A set of implementation-dependent security requirements for a specific product.

Target (ST)	
Target of Evaluation (TOE)	The product under evaluation.
TOE Security Functionality (TSF)	The security functionality of the product under evaluation.
TOE Summary Specification (TSS)	A description of how a TOE satisfies the SFRs in an ST.

1.2.2 Technical Terms

Add-on	Capabilities or functionality added to an application. This term includes plug-ins, extensions, and other controls.
Administrator	The Administrator is responsible for management activities, including setting the policy that is applied by the enterprise on the browser. This administrator is likely to be acting remotely. If the platform is unmanaged by an enterprise, the user can act as the administrator.
Cross-Site Request Forgery (CSRF)	A vulnerability where an attacker gets a target user to execute a script with that user's privileges.
Cross-Site Scripting (XSS)	Injection of untrusted content into a vulnerable web application to render or execute that content on a victim's system.
Domain	A realm of administrative autonomy, authority or control on the internet (e.g., cnn.com).
Extension	A bundle of code added to the browser to add specific functionality that the browser does not provide by default.
HTML5	A new version of HTML that incorporates many new features that enrich the browsing experience.
HyperText Markup Language (HTML)	A language used by web servers to present content to browsers.
HyperText Transfer Protocol (HTTP)	A protocol for communicating on the web.
HyperText Transfer Protocol Secure (HTTPS)	A secure version of HTTP that runs over an encrypted channel (SSL/TLS).
JavaScript	A scripting language commonly integrated into webpages to generate dynamic, interactive content
Mobile Code	Software transmitted from a remote system for execution within a limited execution environment on the local system. Typically, there is no persistent installation and execution begins without the user's consent or even notification. Examples of mobile code technologies include Java applets, Adobe ActionScript, and Microsoft Silverlight. Note that references to mobile code do not refer to JavaScript.
Plug-in	A browser add-on to handle specific types of web content.
Pop-up	A piece of web code that causes a browser to open a window outside the window that is currently in focus.
Port	An application-specific construct that functions as a communications endpoint in a computer's host OS; in a web environment, port 80 is the default port for HTTP communications, although other ports can be used. In a web address, the port follows the domain or sub-domain name (e.g., http://www.cnn.com:80).
Protocol	A system of digital rules for data exchange within or between computers; in a web environment, the typical protocols are HTTP and HTTPS.
Sandbox	A security mechanism for separating running processes, most often used to run untrusted or

	vulnerable processes by reducing their privileges to such an extent that they should not be able to harm the host system.
Sensitive Data	Sensitive data may include all user or enterprise data or may be specific application data such as data transferred to submit a form or complete a transaction. Sensitive data must minimally include personally identifiable information (PII), credentials, and keys. Sensitive data is expected to be identified in the ST.
Sub-domain	An internet domain which is part of a primary domain, denoted by a prefix before the primary domain (e.g., news.cnn.com).
Tabs	A mechanism that allows a browser to display content from multiple websites in the same window.
Web Browser	An application that retrieves and renders content provided by a web server. The terms web browser, browser, and TOE are interchangeable in this document.

1.3 Compliant Targets of Evaluation

The Target of Evaluation (TOE) in this PP-Module is a web browser application running on a desktop or mobile operating system.

Web browsers are client applications that retrieve and render content provided by web servers, primarily using the hypertext transfer protocol (HTTP) or HTTP Secure (HTTPS). Browsers have grown in complexity over the years, starting as tools used to display simple, unchanging websites and becoming sophisticated execution environments for web content. The use of browsers to administer accounts, servers, or embedded systems remotely requires them to handle sensitive information securely. Innovations such as tabs, extensions, and HTML5 have not only increased browser functionality, but also introduced new security concerns. Being the principal method for accessing the internet, and due to their complexity and the information that they process, browsers are a natural target for attackers. As a result, it is paramount that the security of web browsers be improved to reduce the risk to client machines and enterprise networks.

This PP-Module along with the Protection Profile for Application Software [App PP] provide a baseline set of Security Functional Requirements (SFRs) for web browsers running on any operating system regardless of the composition of the underlying platform. The requirements are intended to improve the security of browsers by encouraging the use of operating system security services and requiring the use of sandboxing technologies and environmental mitigations provided by the underlying platform. Additionally, these requirements define security functionality that browsers must provide.

The terms web browser, browser, and TOE are interchangeable in this document.

1.3.1 TOE Boundary

The physical boundary of the web browser is a software application running on a general-purpose operating system. The TOE boundary may include third-party add-ons, but these are non-interfering with respect to security; add-ons provide features that are outside the TOE's logical boundary but must be implemented in such a manner that their inclusion does not compromise the security of the TSF.

1.4 Use Cases

Requirements in this PP-Module are designed to address the security problems in the use cases below. These use cases are intentionally very broad, as web browsers can be used to perform many tasks.

[USE CASE 1] Surfing the Web
Browsers are used to retrieve, display, and render content from the web, such as websites, streaming media, images, and specialized formats (e.g., Java, PDF). They can also be used to write content to websites (web 2.0 – e.g., Facebook). Web surfing can be done over the internet or within an intranet.

[USE CASE 2] Remote Administration Client
Browsers are used to provide remote administration interfaces for systems such as servers, network devices, and embedded systems, to include supervisory control and data acquisition (SCADA) systems, smart TVs, and thermostats. As opposed to surfing the web, where the browser may be interacting with untrusted content, the browser, acting as a Remote Administration Client, is connecting to a server that the user trusts.

[USE CASE 3] Content Creation
Browsers are used to create content via an increasing number of Software as a Service (SaaS) offerings, including Microsoft Office 365, Google Drive, and Adobe Creative Cloud, where user data and records are stored online.

2 Conformance Claims

Conformance Statement

An ST must claim exact conformance to this PP-Module.

The evaluation methods used for evaluating the TOE are a combination of the workunits defined in [CEM] as well as the Evaluation Activities for ensuring that individual SFRs have sufficient supporting evidence in the Security Target and guidance documentation and have been sufficiently tested by the laboratory as part of completing ATE_IND.1. Any functional packages this PP-Module claims similarly contain their own Evaluation Activities that are used in this same manner.

CC Conformance Claims

This PP is conformant to Parts 2 (extended) and 3 (extended) of Common Criteria CC:2022, Revision 1.

PP Claim

This PP-Module does not claim conformance to any other Protection Profile.

No other PPs or PP-Modules are allowed to be specified in a PP-Configuration with this PP-Module beyond its Base-PP.

Package Claim

- This PP-Module is Functional Package for TLS Version 1.1 Conformant.
- This PP-Module is Functional Package for TLS Version 2.0 Conformant.
- This PP-Module conforms to the EAL1 assurance package augmented with ALC_TSU_EXT.1, ASE_OBJ.2, ASE_REQ.2, and ASE_SPD.1.

The functional packages to which the PP-Module conforms include SFRs that are not mandatory to claim for the sake of conformance. An ST that claims one or more of these functional packages may include any non-mandatory SFRs that are appropriate to claim based on the capabilities of the TSF and on any triggers for their inclusion based inherently on the SFR selections made. All security requirements in these packages are intended to satisfy the O.PROTECTED_COMMS TOE security objective of the Base-PP.

3 Security Problem Description

The security problem is described in terms of the threats that the web browser is expected to address, assumptions about the operational environment, and any organizational security policies that it is expected to enforce.

This PP-Module does not repeat the threats, assumptions, and organizational security policies identified in the App PP, though they all apply given the conformance and hence dependence of this PP-Module on it. Together, the threats, assumptions and organizational security policies of the App PP and those defined in this PP-Module describe those addressed by a web browser as the Target of Evaluation.

Notably, browsers are particularly at risk from the T.NETWORK_ATTACK threat identified in the App PP. Attackers can use phishing or another social engineering technique to persuade a user to visit a malicious site. Users may also unintentionally visit malicious sites in the course of web browsing. Such sites then present malicious content to the user's browser to exploit it and perform installation of malware, often with no indication to the user.

3.1 Threats

The following threats are specific to web browsers, and represent an addition to those identified in the App PP.

T.FLAWED_ADDON
Web browser functionality can be extended through the integration of third-party utilities and tools. Malicious or vulnerable add-ons could result in attacks against the system. Such attacks can allow unauthorized access to sensitive information in the browser, unauthorized access to the platform's file system, or privilege escalation that enables unauthorized access to other applications or the operating system.

T.SAME_ORIGIN_VIOLATION
Violating the same-origin policy is a specialized type of network attack (covered generally as T.NETWORK_ATTACK in the App PP), which involves web content violating access control policies enforced by a web browser to separate the content of different web domains. It is specifically identified as a threat to web browsers, since they implement the access control policies that are violated in these attacks.

Attacks which involve same origin violations include:

- Insufficient protection of session tokens can lead to session hijacking, where a token is captured and reused to gain the privileges of the user who initiated the session.
- XSS and CSRF attacks are methods used to compromise user credentials (usually by stealing the user's session token) to a website. These attacks are more likely a result of server security problems, but some browsers incorporate technologies that try to detect the attacks.
- Inadequate sandboxing of browser windows and tabs or a faulty cross domain communications model can lead to leakage of content from one domain in one window or tab to a different domain in a different window or tab. Such attacks leverage the ability of browsers to display content from multiple domains simultaneously.

A malicious actor could implement a malicious website or send a maliciously-crafted URL that an unsuspecting user could open in a vulnerable web browser, which could then be used to harvest sensitive data from a legitimate user or otherwise facilitate impersonation of that user.

3.2 Assumptions

This document does not define any additional assumptions.

3.3 Organizational Security Policies

An organization deploying the TOE is expected to satisfy the organizational security policy listed below in addition to all organizational security policies defined by the claimed Base-PP.

This document does not define any additional OSPs.

4 Security Objectives

This PP-Module adds SFRs to objectives identified in the Base-PP and describes additional objectives specific to this PP-Module.

4.1 Security Objectives for the TOE

O.BROWSER_INTEGRITY
A general version of this objective is defined in the Base-PP. This PP-Module defines a version of the objective that is specific to the functionality that protects the integrity of a web browser application.

O.BROWSER_MANAGEMENT
A general version of this objective is defined in the Base-PP. This PP-Module defines a version of the objective that is specific to the functionality that may be managed by a web browser application.

O.BROWSER_PROTECTED_STORAGE
A general version of this objective is defined in the Base-PP. This PP-Module defines a version of the objective that applies to the data-at-rest protection functionality and considerations that are specific to web browser applications.

O.BROWSER_PROTECTED_COMMS
A general version of this objective is defined in the Base-PP. This PP-Module defines a version of the objective that applies to the data-in-transit protection functionality and considerations that are specific to web browser applications.

O.DOMAIN_ISOLATION
To address the network attack associated with content leakage between different web domains, the browser must ensure that content originating from different domains (e.g., in a tab or iFrame) is properly isolated.

O.ADDON_INTEGRITY
To address issues associated with malicious or flawed add-ons, conformant browsers implement mechanisms to ensure their integrity. This includes verification and validation at installation time and update.

4.2 Security Objectives for the Operational Environment

This PP-Module does not define any objectives for the OE.

No environmental security objectives have been identified that are specific to web browsers. However, any environmental security objectives defined in the Base-PP will also apply to the portion of the TOE that implements web browser functionality.

4.3 Security Objectives Rationale

This section describes how the assumptions, threats, and organizational security policies map to the security objectives.

Table 1: Security Objectives Rationale

Threat, Assumption, or OSP	Security Objectives	Rationale
T.FLAWED_ADDON	O.ADDON_INTEGRITY	The threat T.FLAWED_ADDON is countered by O.ADDON_INTEGRITY, which ensures that a conformant TOE either does not support add-ons at all (in which case there is no possibility of it executing a flawed add-on) or that it supports only add-ons that can prove their integrity.
T.NETWORK_ATTACK (from AppPP)	O.BROWSER_PROTECTED_COMMS	The threat T.NETWORK_ATTACK is countered by O.BROWSER_PROTECTED_COMMS as this provides for the ability of the TOE to resist unauthorized modification via a network vector.
	O.BROWSER_MANAGEMENT	The threat T.NETWORK_ATTACK is countered by O.BROWSER_MANAGEMENT as this provides for the ability to configure the application to defend against network attack.
	O.BROWSER_INTEGRITY	The threat T.NETWORK_ATTACK is countered by O.BROWSER_INTEGRITY as this provides for integrity of transmitted data.
T.NETWORK_EAVESDROP (from AppPP)	O.BROWSER_MANAGEMENT	The threat T.NETWORK_EAVESDROP is countered by O.BROWSER_MANAGEMENT as this provides for the ability to configure the application to protect the confidentiality of its transmitted data.
	O.BROWSER_PROTECTED_	The threat T.NETWORK_EAVESDROP is countered by O.BROWSER_PROTECTED_COMMS as this provides for confidentiality of

	COMMS	transmitted data.
T.PHYSICAL_ ACCESS (from AppPP)	O.BROWSER_ PROTECTED_ STORAGE	The threat T.PHYSICAL_ACCESS is countered by O.BROWSER_PROTECTED_STORAGE, which protects against unauthorized attempts to access physical storage used by the TOE.
T.SAME_ ORIGIN_ VIOLATION	O.DOMAIN_ ISOLATION	The threat T.SAME_ORIGIN_VIOLATION is countered by O.DOMAIN_ISOLATION, which ensures that a conformant TOE will prevent leakage of content between multiple windows or tabs being rendered by the same application.

5 Security Requirements

This chapter describes the security requirements which have to be fulfilled by the product under evaluation. Those requirements comprise functional components from Part 2 and assurance components from Part 3 of [CC]. The following conventions are used for the completion of operations:

- **Refinement** operation (denoted by **bold text** or ~~strikethrough text~~): Is used to add details to a requirement (including replacing an assignment with a more restrictive selection) or to remove part of the requirement that is made irrelevant through the completion of another operation, and thus further restricts a requirement.
- **Selection** (denoted by *italicized text*): Is used to select one or more options provided by the [CC] in stating a requirement.
- **Assignment** operation (denoted by *italicized text*): Is used to assign a specific value to an unspecified parameter, such as the length of a password. Showing the value in square brackets indicates assignment.
- **Iteration** operation: Is indicated by appending the SFR name with a slash and unique identifier suggesting the purpose of the operation, e.g. "/EXAMPLE1."

5.1 App PP Security Functional Requirements Direction

In a PP-Configuration that includes the App PP, the TOE is expected to rely on some of the security functions implemented by the web browser as a whole and evaluated against the App PP. The following sections describe any modifications that the ST author must make to the SFRs defined in the App PP in addition to what is mandated by Section 5.2 TOE Security Functional Requirements.

5.1.1 Modified SFRs

The SFRs listed in this section are defined in the App PP and relevant to the secure operation of the TOE.

5.1.1.1 Cryptographic Support (FCS)

FCS_CKM_EXT.1 Cryptographic Key Generation Services

FCS_CKM_EXT.1.1

The application shall [**selection**:

- *invoke platform-provided functionality for asymmetric key generation*
- *implement asymmetric key generation*

].

Application Note: This SFR is modified from its Base-PP definition to remove the selection for the TOE not requiring asymmetric key generation.

FCS_HTTPS_EXT.1/Client HTTPS Protocol

This SFR is selection-based in the App PP. When the TOE conforms to this PP-Module, this SFR is mandatory because of the modifications that this PP-Module makes to FTP_DIT_EXT.1.

FCS_RBG_EXT.1 Random Bit Generation Services

FCS_RBG_EXT.1.1

The application shall [**selection**:

- *invoke platform-provided DRBG functionality*
- *implement DRBG functionality*

] for its cryptographic operations.

Application Note: This SFR is modified from its Base-PP definition to remove the selection for the TOE using no DRBG functionality.

5.1.1.2 Identification and Authentication (FIA)

FIA_X509_EXT.1 X.509 Certificate Validation

This SFR is selection-based in the App PP. When the TOE conforms to this PP-Module, this SFR is mandatory because of the modifications that this PP-Module makes to FTP_DIT_EXT.1.

FIA_X509_EXT.2 X.509 Certificate Authentication

This SFR is selection-based in the App PP. When the TOE conforms to this PP-Module, this SFR is mandatory because of the modifications that this PP-Module makes to FTP_DIT_EXT.1.

5.1.1.3 Trusted Path/Channels (FTP)

FTP_DIT_EXT.1 Protection of Data in Transit

FTP_DIT_EXT.1.1

The application shall [**selection**:

- *encrypt all transmitted [sensitive data] with [HTTPS as a client in accordance with FCS_HTTPS_EXT.1/Client for [web browsing], TLS as a client as defined in the Functional Package for TLS for [web browsing], DTLS as a client as defined in the Functional Package for TLS for [web browsing]]*
- *invoke platform-provided functionality to encrypt all transmitted sensitive data with [HTTPS, TLS, DTLS] for [web browsing]*

] between itself and another trusted IT product.

Application Note: This SFR is modified from its definition in the App PP to require that the TOE or its platform supports HTTPS, TLS, and DTLS, and that its use of these protocols is only limited to sensitive data. A conformant TOE must support the use of HTTPS, TLS, and DTLS for secure web browsing but is permitted to interact with non-sensitive content over an untrusted channel.

Either the TOE or its platform is permitted to implement TLS and DTLS. If the TOE implements these protocols, FCS_DTLSC_EXT.1, FCS_DTLSC_EXT.2, FCS_TLS_EXT.1, FCS_TLSC_EXT.1, and FCS_TLSC_EXT.2 from the TLS package must be claimed at minimum because a web browser is required to support mutually-authenticated TLS and DTLS.

5.2 TOE Security Functional Requirements

The following section describes the SFRs that must be satisfied by any TOE that claims conformance to this PP-Module. These SFRs must be claimed regardless of which PP-Configuration is used to define the TOE.

5.2.1 User Data Protection (FDP)

FDP_ACF_EXT.1 Local and Session Storage Separation

FDP_ACF_EXT.1.1

The TSF shall separate local (permanent) and session (ephemeral) storage based on domain, protocol, and port:

- Session storage shall be accessible only from the originating window or tab;
- Local storage shall only be accessible from windows or tabs running the same web application.

Application Note: The separation of local and session storage is described in World Wide Web Consortium (W3C) Proposed Recommendation: "Web Storage."

FDP_COO_EXT.1 Cookie Blocking

FDP_COO_EXT.1.1

The TSF shall provide the capability to block the storage of third-party cookies by websites.

FDP_SBX_EXT.1 Sandboxing of Rendering Processes

FDP_SBX_EXT.1.1

The TSF shall ensure that webpage rendering is performed in a process that is restricted in the following manner:

- The rendering process can only directly access the area of the file system dedicated to the browser.
- The rendering process can only directly invoke inter-process communication mechanisms with its own browser processes.
- The rendering process has reduced privilege with respect to other browser processes [**selection, choose one of**: *[**assignment**: other methods by which the principle of least privilege is implemented for rendering processes], in no other ways*].

Application Note: Web browsers implement a variety of methods to ensure that the process that renders HTML and interprets JavaScript operates in a constrained environment in order to reduce the risk that the rendering process can be corrupted by the HTML or JavaScript it is processing. This component requires the browser to lower the privileges of rendering processes by ensuring that it cannot directly access the file system of the host, and that it cannot use inter-process communication (IPC) mechanisms provided by the host to communicate with non-browser processes on the host. Typically, if a rendering process needs to access a file or communicate with a non-browser process, it must request such access through the TSF (which is allowed by the requirement).

In addition to the two required measures, other measures can be implemented depending on the browser and the host platform. These may involve such actions as changing the owner of the rendering process to a low-privileged account or dropping platform-defined privileges in the rendering process. The ST author fills in the additional measures implemented by the browser.

FDP_SOP_EXT.1 Same Origin Policy

FDP_SOP_EXT.1.1

The TSF shall only permit scripts contained in one webpage to access data in a second webpage if both pages are from the same origin.

FDP_SOP_EXT.1.2

The TSF shall enforce the same origin policy for all domains.

Application Note: The Same Origin Policy concept is described in RFC 6454, "The Web Origin Concept." Origin is defined as the combination of domain, protocol, and port. Two URIs sharing the same domain, protocol, and port are considered to have the same origin.

FDP_STR_EXT.1 Secure Transmission of Cookie Data

FDP_STR_EXT.1.1

The TSF shall ensure that cookies containing the 'secure' attribute in the set-cookie header are sent over HTTPS.

Application Note: The set-cookie header functionality is described in RFC 6265, "HTTP State Management Mechanism."

FDP_TRK_EXT.1 Tracking Information Collection

FDP_TRK_EXT.1.1

The TSF shall provide notification to the user when tracking information for [**selection**:

- *geolocation*
- *browser history*
- *browser preferences*
- *browser statistics*

] is requested by a website.

5.2.2 Security Management (FMT)

FMT_MOF_EXT.1 Management of Functions Behavior

FMT_MOF_EXT.1.1

The TSF shall be capable of performing the following management functions, controlled by the administrator or user as shown:

- M = Mandatory
- O = Optional

#	Management Function	Administrator	User
1	Enable and disable storage of third-party cookies	O	M
2	Enable and disable use of OCSP for obtaining the revocation status of X.509 certificates	O	O
3	Configure inclusion of user-agent information in HTTP headers	O	O
4	Enable and disable ability for websites to collect tracking information about the user through [**selection**: *zombie cookies, add-on based tracking (e.g., Flash cookies), browsing history, [**assignment**: other tracking mechanisms]*]	O	O
5	Enable and disable deletion of stored browsing data (cache, web form information)	O	M
6	Enable and disable storage of sensitive information (e.g., auto-fill, auto-complete) in persistent storage	O	O
7	Configure cookie cache size	O	O
8	Configure cache size	O	O
9	Enable and disable interaction with Graphic Processing Units (GPUs)	O	O
10	Configure the ability to advance to a website with an invalid or unvalidated X.509 certificate	O	O
11	Enable and disable establishment of a trusted channel if the browser cannot establish a connection to determine the validity of a	O	O

	certificate		
12	Configure the use of an application reputation service to detect malicious applications prior to download	O	O
13	Configure the use of a URL reputation service to detect sites that contain malware or phishing content	O	O
14	Enable and disable automatic installation of software updates and patches	O	O
15	Enable and disable ability for websites to register protocol handlers	O	O
16	Enable and disable display notification when unsigned, untrusted, or unverified mobile code is encountered	O	O
17	Enable and disable user's ability to select default actions upon download of a file (e.g., always open or always save a downloaded file)	O	O
18	Enable and disable launching of downloaded files outside the browser	O	O
19	Enable and disable JavaScript	O	O
20	Enable and disable [**selection**: *ActiveX*, *Flash*, *Java*, *[**assignment**: other mobile code types supported by the browser]*] mobile code	O	O
21	Enable and disable support for add-ons	O	O
22	Enable and disable individual add-ons	O	O
23	Enable and disable HSTS mode	O	O

Application Note: For these management functions, the term "Administrator" refers to the administrator of a non-mobile device or the device owner of a mobile device. The intent of this requirement is to allow the user and administrator of the platform to configure the browser with configuration policies. If the administrator has not set a policy for a particular function, the user may still perform that function. Enforcement of the policy is done by the browser itself or the browser and its platform in coordination with each other. Disabling OCSP is only permitted if "CRL" was selected in FIA_X509_EXT.1.1 (in App PP).

5.2.3 Protection of the TSF (FPT)

FPT_AON_EXT.1 Support for Only Trusted Add-ons

FPT_AON_EXT.1.1

The TSF shall include the capability to load [**selection, choose one of**: *trusted add-ons*, *no add-ons*].

Application Note: If trusted add-ons is selected in FPT_AON_EXT.1.1, the TOE must also claim the selection-based SFR FPT_AON_EXT.2.

If the browser does not include support for installing only trusted add-ons, this requirement can be met by demonstrating the ability to disable all support for add-ons as specified in FMT_MOF_EXT.1.

FPT_DNL_EXT.1 File Downloads

FPT_DNL_EXT.1.1

The TSF shall prevent downloaded content from launching automatically.

FPT_DNL_EXT.1.2

The TSF shall present the user with the option to either save or discard downloaded files.

Application Note: This requirement ensures that if the user intentionally (via clicking on a link) or unintentionally initiates the download of a file, the browser will intervene by, for example, opening a dialog box that presents the user with the option either to save the file to the file system download the file. In this context, an executable is a file containing code for a software program that is invoked independent of and outside the context of the browser. It does not include mobile code, scripts, or add-ons.

FPT_MCD_EXT.1 Mobile Code

FPT_MCD_EXT.1.1

The TSF shall support the capability to execute [**selection, choose one of**:

- *signed [**selection**:*
  - *ActiveX*
  - *Flash*
  - *Java*
  - *ActionScript*
  - *[**assignment**: other mobile code types supported by the browser]*

  ]
- *no*

] mobile code.

FPT_MCD_EXT.1.2

The TSF shall [**selection, choose one of**: *automatically discard, provide the user with the option to discard*] unsigned, untrusted, or unverified [**selection**:

- *ActiveX*
- *Flash*
- *Java*
- *ActionScript*
- *[**assignment**: other mobile code types supported by the browser]*

] mobile code without executing it.

Application Note: The ST author must specify all mobile code types for which the browser provides this support.

An authorized signer may directly sign the code itself, or the code may be delivered over an authenticated HTTPS connection with an authorized entity.

5.3 TOE Security Functional Requirements Rationale

The following rationale provides justification for each security objective for the TOE, showing that the SFRs are suitable to meet and achieve the security objectives:

Table 2: SFR Rationale

Objective	Addressed by	Rationale
O.BROWSER_INTEGRITY	FPT_DNL_EXT.1	FPT_DNL_EXT.1 supports the objective by preventing the automatic execution of downloaded files which could otherwise cause integrity violations to the TOE itself or to its platform.
	FPT_MCD_EXT.1	FPT_MCD_EXT.1 supports the objective by preventing the automatic execution of mobile code which could otherwise cause integrity violations to the TOE itself or to its platform.
	FPT_INT_EXT.1 (objective)	FPT_INT_EXT.1 supports the objective by optionally requiring the TSF to implement a reputation service to prevent the acquisition of potentially malicious applications.
O.BROWSER_MANAGEMENT	FDP_TRK_EXT.1	FDP_TRK_EXT.1 supports the objective by notifying the user when various data is being tracked to allow for control of the disclosure of configuration information.
	FMT_MOF_EXT.1	FMT_MOF_EXT.1 supports the objective by defining the management functionality that is specific to web browser applications.
O.BROWSER_PROTECTED_STORAGE	FDP_COO_EXT.1	FDP_COO_EXT.1 supports the objective by defining a mechanism to prevent untrusted data from being loaded into protected storage.
	FDP_PST_EXT.1 (optional)	FDP_PST_EXT.1 supports the objective by optionally defining the minimum set of persistent data that the TSF is required to store.
	FPT_INT_EXT.1 (objective)	FPT_INT_EXT.1 supports the objective by optionally requiring the TSF to implement a mechanism to protect against downloading known malicious applications that may adversely affect data stored at rest.
O.BROWSER_PROTECTED_COMMS	FCS_CKM_EXT.1 (modified from Base-PP)	FCS_CKM_EXT.1 supports the objective by requiring that the TSF provide or invoke a cryptographic function for asymmetric key generation.
	FCS_HTTPS_EXT.1/Client	FCS_HTTPS_EXT.1/Client supports the objective by defining

	(from Base-PP)	the TSF's implementation of HTTPS.
	FCS_RBG_EXT.1 (modified from Base-PP)	FCS_RBG_EXT.1 supports the objective by requiring that the TSF provide or invoke a DRBG for secure key generation.
	FIA_X509_EXT.1 (from Base-PP)	FIA_X509_EXT.1 supports the objective by requiring the TSF to implement or invoke an X.509 certificate validation service.
	FIA_X509_EXT.2 (from Base-PP)	FIA_X509_EXT.2 supports the objective by defining the TOE's use of X.509 certificates and what behavior the TOE takes when the revocation status of a certificate cannot be determined.
	FTP_DIT_EXT.1 (modified from Base-PP)	FTP_DIT_EXT.1 supports the objective by specifying the trusted communications channels used by the TOE to protect data in transit.
	FDP_STR_EXT.1	FDP_STR_EXT.1 supports the objective by requiring the use of HTTPS for certain types of data transfer.
	FCS_STS_EXT.1 (objective)	FCS_STS_EXT.1 supports the objective by optionally requiring the TSF to implement HSTS for secure data transmission.
	FPT_INT_EXT.2 (objeictve)	FPT_INT_EXT.2 supports the objective by optionally requiring the TSF to implement a URL reputation service that can block communications with malicious entities.
O.DOMAIN_ ISOLATION	FDP_ACF_EXT.1	FDP_ACF_EXT.1 supports the objective by isolating local and session storage to its origin point.
	FDP_SBX_EXT.1	FDP_SBX_EXT.1 supports the objective by ensuring that rendering of content is isolated to its origin point.
	FDP_SOP_EXT.1	FDP_SOP_EXT.1 supports the objective by enforcing the concept of a same origin policy to prevent web content with different origins from interacting with one another.
O.ADDON_ INTEGRITY	FPT_AON_EXT.1	FPT_AON_EXT.1 supports the objective by specifying whether the TSF has the ability to load add-ons.
	FPT_AON_EXT.2 (selection-based)	FPT_AON_EXT.2 supports the objective by defining a cryptographic method for the TSF to validate the integrity of add-ons if the TOE supports their use.

6 Consistency Rationale

6.1 Protection Profile for web browsers

6.1.1 Consistency of TOE Type

If this PP-Module is used to extend the App PP, the TOE type for the overall TOE is still a software application. The TOE boundary is simply extended to include the web browser functionality that is built into the application so that additional security functionality is claimed within the scope of the TOE.

The only asset for the TOE is the software executable and sensitive data that comprises the TOE. The entire TOE as defined by the combination of the Base-PP and this PP-Module is a single asset. The only differences to the threat model are that the PP-Module introduces the concept of add-ons, which introduces the threat of an add-on being flawed in some way, and that the PP-Module introduces the concept of a same-origin violation, which occurs through a use case specific to web browser applications.

6.1.2 Consistency of Security Problem Definition

Listed below are the threats, objectives, and OSPs defined in this PP-Module with rationale for their consistency with the App PP. The PP-Module shares the executable application asset with the App PP but defines additional threats because the PP-Module defines a specific type of software application with potential exploits that are common to the application type.

Note that the PP-Module is implicitly consistent with any claimed functional packages because the applicable functional packages do not have security problem definitions of their own; per section 2, any claimed functional package is intended to support the O.PROTECTED_COMMS objective in the App PP, which helps mitigate the T.NETWORK_ATTACK and T.NETWORK_EAVESDROP threats in that PP.

PP-Module Threat, Assumption, OSP	Consistency Rationale
T.FLAWED_ADDON	The threat of a user installing a flawed add-on is consistent with the T.LOCAL_ATTACK threat from the Base-PP. A flawed add-on, whether crafted deliberately or unintentionally, could cause the product to operate in a manner where it or its platform can be compromised.
T.SAME_ORIGIN_VIOLATION	This threat extends the security problem definition of the Base-PP by defining a potential vulnerability that specifically applies to the content that is handled by web browsers.

6.1.3 Consistency of Objectives

Listed below are the security objectives defined in this PP-Module with rationale for their consistency with the App PP. The PP-Module shares the executable application asset with the App PP but defines additional security objectives because the PP-Module defines a specific type of software application with security functionality that is common to the application type.

Note that the PP-Module is implicitly consistent with any claimed functional packages because the applicable functional packages do not have TOE objecitves of their own; per section 2, any claimed functional package is intended to support the O.PROTECTED_COMMS objective in the App PP. The objectives for the TOEs are consistent with the App PP based on the following rationale:

PP-Module TOE Objective	Consistency Rationale
O.BROWSER_INTEGRITY	This objective is an enhancement to the O.INTEGRITY objective defined in the Base-PP, specifically in regards to the integrity protection mechanisms that apply to web browsers.
O.BROWSER_MANAGEMENT	This objective is an enhancement to the O.MANAGEMENT objective defined in the Base-PP, specifically in regards to the secure administration of functions that are particular to web browser applications.
O.BROWSER_PROTECTED_STORAGE	This objective is an enhancement to the O.PROTECTED_STORAGE objective defined in the Base-PP, specifically in regards to the data-at-rest protection that applies to web browser applications.
O.BROWSER_PROTECTED_COMMS	This objective is an enhancement to the O.PROTECTED_COMMS objective defined in the Base-PP, specifically in regards to the data-in-transit protection that applies to web browser applications.
O.DOMAIN_ISOLATION	This objective applies to functionality that is specific to web browser applications and is therefore beyond the original scope of the Base-PP.
O.ADDON_INTEGRITY	This objective is an enhancement to the O.BROWSER_INTEGRITY objective defined in the Base-PP. Where O.BROWSER_INTEGRITY is concerned with the integrity of the TOE application,

	O.ADDON_INTEGRITY is concerned with the integrity of third-party add-ons that can be loaded into the TOE.

This PP-Module does not define any objectives for the TOE's operational environment.

6.1.4 Consistency of Requirements

This PP-Module identifies several SFRs from the App PP that are needed to support web browser functionality. This is considered to be consistent because the functionality provided by the App PP is being used for its intended purpose. The PP-Module also identifies a number of modified SFRs from the App PP that are used entirely to provide functionality for web browsers. The rationale for why this does not conflict with the claims defined by the App PP are as follows:

PP-Module Requirement	Consistency Rationale
Modified SFRs	
FCS_CKM_EXT.1	This SFR is changed from its definition in the App PP to remove one of the available selection options because it will never apply in the case where the TOE conforms to this PP-Module.
FCS_HTTPS_EXT.1/Client	This SFR is unchanged from its definition in the App PP; the SFR is recategorized from selection-based to mandatory when the TOE conforms to this PP-Module.
FCS_RBG_EXT.1	This SFR is changed from its definition in the App PP to remove one of the available selection options because it will never apply in the case where the TOE conforms to this PP-Module.
FIA_X509_EXT.1	This SFR is unchanged from its definition in the App PP; the SFR is recategorized from selection-based to mandatory when the TOE conforms to this PP-Module.
FIA_X509_EXT.2	This SFR is unchanged from its definition in the App PP; the SFR is recategorized from selection-based to mandatory when the TOE conforms to this PP-Module.
FTP_DIT_EXT.1	This SFR is changed from its definition in the App PP to mandate the protection of sensitive data using only specified protocols.
Additional SFRs	
This PP-Module does not add any requirements when the App PP is the base.	
Mandatory SFRs	
FDP_ACF_EXT.1	This SFR defines domain separation of web content when a web browser is simultaneously accessing content from multiple sources. It does not impact the App PP functionality.
FDP_COO_EXT.1	This SFR defines behavior for handling cookies, which are data specific to web browser applications. It does not impact the App PP functionality.
FDP_SBX_EXT.1	This SFR defines behavior for rendering of webpages, which is by definition functionality that is associated with web browser applications. It does not impact the App PP functionality.
FDP_SOP_EXT.1	This SFR defines behavior for script execution on webpages, which is by definition functionality that is associated with web browser applications. It does not impact the App PP functionality.
FDP_STR_EXT.1	This SFR defines behavior for handling cookies, which are data specific to web browser applications. It does not impact the App PP functionality.
FDP_TRK_EXT.1	This SFR defines behavior for handling tracking information that is specific to web browser applications. It does not impact the App PP functionality.
FMT_MOF_EXT.1	This SFR defines a specific set of management functions for a web browser. It does not impact the App PP functionality.
FPT_AON_EXT.1	This SFR defines what types of add-ons a web browser may use. It does not impact the App PP functionality.
FPT_DNL_EXT.1	This SFR defines behavior for handling file data that can be downloaded by a web browser. It does not impact the App PP functionality.
FPT_MCD_EXT.1	This SFR defines behavior for mobile code that is rendered by a web browser. It does not impact the App PP functionality.
Optional SFRs	

FDP_PST_EXT.1	This SFR defines the persistent information that must be stored for web browser functionality to work as intended. It does not impact functionality described by the App PP.
Objective SFRs	
FCS_STS_EXT.1	This SFR defines behavior for implementation of HSTS, which is a communications mechanism specific to web content. It does not impact the App PP functionality.
FPT_INT_EXT.1	This SFR defines behavior interaction with a reputation service for file data that the TOE can be used to download. It does not impact the App PP functionality.
FPT_INT_EXT.2	This SFR defines behavior interaction with a reputation service for web content that the TOE can be used to interact with. It does not impact the App PP functionality.
Implementation-based SFRs	
This PP-Module does not define any Implementation-based requirements.	
Selection-based SFRs	
FPT_AON_EXT.2	This SFR defines how web browsers verify add-ons. It does not impact functionality described by the App PP.

Appendix A - Optional SFRs

A.1 Strictly Optional Requirements

A.1.1 User Data Protection (FDP)

FDP_PST_EXT.1 Storage of Persistent Information

FDP_PST_EXT.1.1

The TSF shall provide the capability to operate without storing persistent data to the file system with the following exceptions: [**selection**: *credential information*, *administrator-provided configuration information*, *certificate revocation information*, *no exceptions*].

Application Note: Any data that persists after the browser closes, including temporary files, is considered to be persistent data.

A.2 Objective Requirements

A.2.1 Cryptographic Support (FCS)

FCS_STS_EXT.1 Strict Transport Security

FCS_STS_EXT.1.1

The TSF shall implement HTTP Strict-Transport-Security according to RFC 6797.

FCS_STS_EXT.1.2

The TSF shall retain persistent data signaling HSTS enablement for the time span declared by the website in a max-age directive.

FCS_STS_EXT.1.3

The TSF shall cache the "freshest" Strict Security policy information.

Application Note: Freshness refers to the length of time between generation by the origin server and the expiration time when the origin server specifies that a stored response can no longer be used by a cache without further validation (RFCs 6797 and 7234).

A.2.2 Protection of the TSF (FPT)

FPT_INT_EXT.1 Interactions with Application Reputation Services

FPT_INT_EXT.1.1

The TSF shall use an application reputation service to prevent downloading of malicious applications.

Application Note: An application reputation service is an online service that identifies malicious applications; it is used to detect such applications prior to downloading them. Using a reputation service would require configuration of the trusted service to be used. The quality of the reputation service may fall outside the scope of the evaluation.

FPT_INT_EXT.2 Interactions with URL Reputation Services

FPT_INT_EXT.2.1

The TSF shall use a URL reputation service to prevent connections with malicious websites.

Application Note: A URL reputation service is an online service that identifies websites with malicious or phishing content applications; it is used to detect such websites prior to allowing users to access them. The goal of this requirement is to ensure that the browser is prevented from establishing connections with known-bad sources of malware on the internet. The specifics of the sequence of actions taken before a block decision is made may depend upon the specific implementation of the browser. For example, some browsers might implement the check for malicious content by checking against the list of bad URLs provided by the URL reputation service in real time. Other browsers may download updated lists of bad URLs at browser startup, updating the list periodically from the URL reputation service or services until the browser is terminated. Ultimately, the result should be that the browser blocks the connection to the bad URL.

A.3 Implementation-based Requirements

This PP-Module does not define any Implementation-based SFRs.

Appendix B - Selection-based Requirements

B.1 Protection of the TSF (FPT)

FPT_AON_EXT.2 Trusted Installation and Update for Add-ons

The inclusion of this selection-based component depends upon selection in FPT_AON_EXT.1.1.

FPT_AON_EXT.2.1

The TSF shall [**selection**: *provide the ability, leverage the platform*] to provide a means to cryptographically verify add-ons using a digital signature mechanism and [**selection, choose one of**: *published hash, no other functions*] prior to installation and update.

FPT_AON_EXT.2.2

The TSF shall [**selection**: *provide the ability, leverage the platform*] to query the current version of the add-on.

FPT_AON_EXT.2.3

The TSF shall prevent the automatic installation of add-ons.

Application Note: This selection-based SFR is claimed when trusted add-ons is selected in FPT_AON_EXT.1.1.

Appendix C - Extended Component Definitions

This appendix contains the definitions for all extended requirements specified in the PP-Module.

C.1 Extended Components Table

All extended components specified in the PP-Module are listed in this table:

Table 3: Extended Component Definitions

Functional Class	Functional Components
Cryptographic Support (FCS)	FCS_STS_EXT Strict Transport Security
Protection of the TSF (FPT)	FPT_AON_EXT Add-Ons FPT_DNL_EXT File Downloads FPT_INT_EXT Reputation Service Interaction FPT_MCD_EXT Mobile Code
Security Management (FMT)	FMT_MOF_EXT Management of Functions Behavior
User Data Protection (FDP)	FDP_ACF_EXT Access Control Functions FDP_COO_EXT Cookie Blocking FDP_PST_EXT Storage of Persistent Information FDP_SBX_EXT Sandboxing FDP_SOP_EXT Same Origin Policy FDP_STR_EXT Secure Transmission of Cookie Data FDP_TRK_EXT Tracking Information Collection

C.2 Extended Component Definitions

C.2.1 Cryptographic Support (FCS)

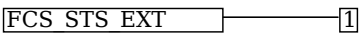
This PP-Module defines the following extended components as part of the FCS class originally defined by CC Part 2:

C.2.1.1 FCS_STS_EXT Strict Transport Security

Family Behavior

Components in this family define requirements for the implementation of HTTP Strict-Transport-Security.

Component Leveling



FCS_STS_EXT.1, Strict Transport Security, requires the TSF to implement HTTP Strict-Transport-Security.

Management: FCS_STS_EXT.1

The following actions could be considered for the management functions in FMT:

- Enable and disable HSTS mode.

Audit: FCS_STS_EXT.1

There are no auditable events foreseen.

FCS_STS_EXT.1 Strict Transport Security

Hierarchical to: No other components.

Dependencies to: FCS_HTTPS_EXT.1 HTTPS Protocol

FCS_STS_EXT.1.1

The TSF shall implement HTTP Strict-Transport-Security according to RFC 6797.

FCS_STS_EXT.1.2

The TSF shall retain persistent data signaling HSTS enablement for the time span declared by the website in a max-age directive.

FCS_STS_EXT.1.3

The TSF shall cache the "freshest" Strict Security policy information.

C.2.2 Protection of the TSF (FPT)

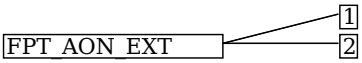
This PP-Module defines the following extended components as part of the FPT class originally defined by CC Part 2:

C.2.2.1 FPT_AON_EXT Add-Ons

Family Behavior

Components in this family define requirements for the secure handling of add-ons that can be installed on top of the TOE.

Component Leveling



FPT_AON_EXT.1, Support for Only Trusted Add-ons, requires the TSF to either support no add-ons or to only support trusted add-ons.

FPT_AON_EXT.2, Trusted Installation and Update for Add-ons, requires the TSF to implement a method to verify the integrity of add-ons and ensure that untrusted or unknown add-ons are not loaded for use.

Management: FPT_AON_EXT.1

The following actions could be considered for the management functions in FMT:

- Enable and disable support for add-ons.

Audit: FPT_AON_EXT.1

There are no auditable events foreseen.

FPT_AON_EXT.1 Support for Only Trusted Add-ons

Hierarchical to:	No other components.
Dependencies to:	No dependencies.

FPT_AON_EXT.1.1

The TSF shall include the capability to load [**selection, choose one of**: *trusted add-ons*, *no add-ons*].

Management: FPT_AON_EXT.2

No specific management functions are identified.

Audit: FPT_AON_EXT.2

There are no auditable events foreseen.

FPT_AON_EXT.2 Trusted Installation and Update for Add-ons

Hierarchical to:	No other components.
Dependencies to:	FCS_COP.1 Cryptographic Operation FPT_AON_EXT.1 Support for Only Trusted Add-Ons

FPT_AON_EXT.2.1

The TSF shall [**selection**: *provide the ability*, *leverage the platform*] to provide a means to cryptographically verify add-ons using a digital signature mechanism and [**selection, choose one of**: *published hash*, *no other functions*] prior to installation and update.

FPT_AON_EXT.2.2

The TSF shall [**selection**: *provide the ability*, *leverage the platform*] to query the current version of the add-on.

FPT_AON_EXT.2.3

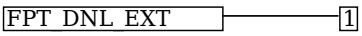
The TSF shall prevent the automatic installation of add-ons.

C.2.2.2 FPT_DNL_EXT File Downloads

Family Behavior

Components in this family define requirements for downloaded content.

Component Leveling



FPT_DNL_EXT.1, File Downloads, requires the TSF to intervene in the case it is prompted to download executable file data.

Management: FPT_DNL_EXT.1

The following actions could be considered for the management functions in FMT:

- Enable and disable user's ability to select default actions upon download of a file (e.g., always open or always save a downloaded file).
- Enable and disable launching of downloaded files outside the browser.

Audit: FPT_DNL_EXT.1

There are no auditable events foreseen.

FPT_DNL_EXT.1 File Downloads

Hierarchical to: No other components.

Dependencies to: No dependencies.

FPT_DNL_EXT.1.1

The TSF shall prevent downloaded content from launching automatically.

FPT_DNL_EXT.1.2

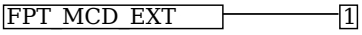
The TSF shall present the user with the option to either save or discard downloaded files.

C.2.2.3 FPT_MCD_EXT Mobile Code

Family Behavior

Components in this family define requirements for execution of mobile code.

Component Leveling



FPT_MCD_EXT.1, Mobile Code, requires the TSF to identify the mobile code types it supports and to ensure that a mechanism exists to prevent the automatic execution of potentially malicious mobile code.

Management: FPT_MCD_EXT.1

The following actions could be considered for the management functions in FMT:

- Enable and disable display notification when unsigned, untrusted, or unverified mobile code is encountered.
- Enable and disable support for mobile code.

Audit: FPT_MCD_EXT.1

There are no auditable events foreseen.

FPT_MCD_EXT.1 Mobile Code

Hierarchical to: No other components.

Dependencies to: No dependencies.

FPT_MCD_EXT.1.1

The TSF shall support the capability to execute [**selection, choose one of**: *signed [**assignment**: supported mobile code types] , no*] mobile code.

FPT_MCD_EXT.1.2

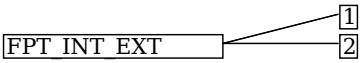
The TSF shall [**selection, choose one of**: *automatically discard, provide the user with the option to discard*] unsigned, untrusted, or unverified [**assignment**: *supported mobile code types*] mobile code without executing it.

C.2.2.4 FPT_INT_EXT Reputation Service Interaction

Family Behavior

Components in this family define requirements for the TOE's interaction with reputation services that can provide an assessment of the trustworthiness of data presented to the TSF.

Component Leveling



FPT_INT_EXT.1, Interactions with Application Reputation Services, requires the TSF to be able to interact with an application reputation service to assess whether application data is potentially malicious.

FPT_INT_EXT.2, Interactions with URL Reputation Services, requires the TSF to be able to interact with a URL reputation service to assess whether websites are potentially malicious.

Management: FPT_INT_EXT.1

The following actions could be considered for the management functions in FMT:

- Configure the use of an application reputation service to detect malicious applications prior to download.

Audit: FPT_INT_EXT.1

There are no auditable events foreseen.

FPT_INT_EXT.1 Interactions with Application Reputation Services

Hierarchical to: No other components.

Dependencies to: No dependencies.

FPT_INT_EXT.1.1

The TSF shall use an application reputation service to prevent downloading of malicious applications.

Management: FPT_INT_EXT.2

The following actions could be considered for the management functions in FMT:

- Configure the use of a URL reputation service to detect sites that contain malware or phishing content.

Audit: FPT_INT_EXT.2

There are no auditable events foreseen.

FPT_INT_EXT.2 Interactions with URL Reputation Services

Hierarchical to: No other components.

Dependencies to: No dependencies.

FPT_INT_EXT.2.1

The TSF shall use a URL reputation service to prevent connections with malicious websites.

C.2.3 Security Management (FMT)

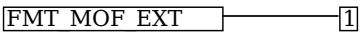
This PP-Module defines the following extended components as part of the FMT class originally defined by CC Part 2:

C.2.3.1 FMT_MOF_EXT Management of Functions Behavior

Family Behavior

Components in this family define requirements for technology-specific management functions that are not enumerated in the Part 2 family FMT_MOF.

Component Leveling



FMT_MOF_EXT.1, Management of Functions Behavior, requires the TSF to implement management functions specified in the SFR.

Management: FMT_MOF_EXT.1

No specific management functions are identified.

Audit: FMT_MOF_EXT.1

There are no auditable events foreseen.

FMT_MOF_EXT.1 Management of Functions Behavior

Hierarchical to: No other components.

Dependencies to: No dependencies.

FMT_MOF_EXT.1.1

The TSF shall be capable of performing the following management functions, controlled by the administrator or user as shown: [**assignment**: *list of management functions to be performed by role*].

C.2.4 User Data Protection (FDP)

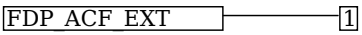
This PP-Module defines the following extended components as part of the FDP class originally defined by CC Part 2:

C.2.4.1 FDP_ACF_EXT Access Control Functions

Family Behavior

Components in this family define requirements for data access control beyond those which are specified in the Part 2 family FDP_ACF.

Component Leveling



FDP_ACF_EXT.1, Local and Session Storage Separation, requires the TSF to enforce data protection mechanisms such that user data is only accessible from its originator.

Management: FDP_ACF_EXT.1

No specific management functions are identified.

Audit: FDP_ACF_EXT.1

There are no auditable events foreseen.

FDP_ACF_EXT.1 Local and Session Storage Separation

Hierarchical to: No other components.

Dependencies to: No dependencies.

FDP_ACF_EXT.1.1

The TSF shall separate local (permanent) and session (ephemeral) storage based on domain, protocol, and port:

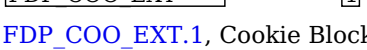
- Session storage shall be accessible only from the originating window or tab;
- Local storage shall only be accessible from windows or tabs running the same web application.

C.2.4.2 FDP_COO_EXT Cookie Blocking

Family Behavior

Components in this family define requirements for controlling whether the TOE stores third-party cookie data.

Component Leveling



FDP_COO_EXT.1, Cookie Blocking, requires the TSF to have a configurable mechanism for blocking the storage of third-party cookies.

Management: FDP_COO_EXT.1

The following actions could be considered for the management functions in FMT:

- Enable and disable storage of third-party cookies.

Audit: FDP_COO_EXT.1

There are no auditable events foreseen.

FDP_COO_EXT.1 Cookie Blocking

Hierarchical to: No other components.

Dependencies to: No dependencies.

FDP_COO_EXT.1.1

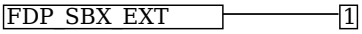
The TSF shall provide the capability to block the storage of third-party cookies by websites.

C.2.4.3 FDP_SBX_EXT Sandboxing

Family Behavior

Components in this family define requirements for ensuring domain separation through sandboxing.

Component Leveling



FDP_SBX_EXT.1, Sandboxing of Rendering Processes, requires the TSF to implement sandboxing of rendering processes such that least privilege is enforced on the rendering process.

Management: FDP_SBX_EXT.1

No specific management functions are identified.

Audit: FDP_SBX_EXT.1

There are no auditable events foreseen.

FDP_SBX_EXT.1 Sandboxing of Rendering Processes

Hierarchical to: No other components.

Dependencies to: No dependencies.

FDP_SBX_EXT.1.1

The TSF shall ensure that webpage rendering is performed in a process that is restricted in the following manner:

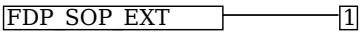
- The rendering process can only directly access the area of the file system dedicated to the browser.
- The rendering process can only directly invoke inter-process communication mechanisms with its own browser processes.
- The rendering process has reduced privilege with respect to other browser processes [**selection, choose one of**: *[**assignment**: other methods by which the principle of least privilege is implemented for rendering processes], in no other ways*].

C.2.4.4 FDP_SOP_EXT Same Origin Policy

Family Behavior

Components in this family define requirements for implementation of the Same Origin Policy concept.

Component Leveling



FDP_SOP_EXT.1, Same Origin Policy, requires the TSF to implement the Same Origin Policy concept for web content.

Management: FDP_SOP_EXT.1

No specific management functions are identified.

Audit: FDP_SOP_EXT.1

There are no auditable events foreseen.

FDP_SOP_EXT.1 Same Origin Policy

Hierarchical to: No other components.

Dependencies to: No dependencies.

FDP_SOP_EXT.1.1

The TSF shall only permit scripts contained in one webpage to access data in a second webpage if both pages are from the same origin.

FDP_SOP_EXT.1.2

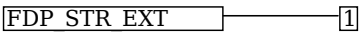
The TSF shall enforce the same origin policy for all domains.

C.2.4.5 FDP_STR_EXT Secure Transmission of Cookie Data

Family Behavior

Components in this family define requirements for using HTTPS to transmit sensitive cookie data.

Component Leveling



FDP_STR_EXT.1, Secure Transmission of Cookie Data, requires the TSF to use HTTPS to transmit cookie data that has a security-relevant attribute.

Management: FDP_STR_EXT.1

No specific management functions are identified.

Audit: FDP_STR_EXT.1

There are no auditable events foreseen.

FDP_STR_EXT.1 Secure Transmission of Cookie Data

Hierarchical to: No other components.

Dependencies to: FCS_HTTPS_EXT.1 HTTPS Protocol

FDP_STR_EXT.1.1

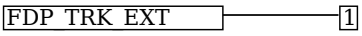
The TSF shall ensure that cookies containing the 'secure' attribute in the set-cookie header are sent over HTTPS.

C.2.4.6 FDP_TRK_EXT Tracking Information Collection

Family Behavior

Components in this family define requirements for notifying a user when certain data that reflects the usage of the TOE is being tracked.

Component Leveling



FDP_TRK_EXT.1, Tracking Information Collection, requires the TSF to specify the data tracking that results in user notification.

Management: FDP_TRK_EXT.1

The following actions could be considered for the management functions in FMT:

- Enable and disable ability for websites to collect tracking information about the user.
- Enable and disable deletion of stored browsing data.

Audit: FDP_TRK_EXT.1

There are no auditable events foreseen.

FDP_TRK_EXT.1 Tracking Information Collection

Hierarchical to: No other components.

Dependencies to: No dependencies.

FDP_TRK_EXT.1.1

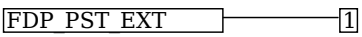
The TSF shall provide notification to the user when tracking information for [**assignment**: *list of trackable browser data*] is requested by a website.

C.2.4.7 FDP_PST_EXT Storage of Persistent Information

Family Behavior

Components in this family define requirements for the minimum amount of information that may be stored persistently by the TSF while retaining its functionality.

Component Leveling



FDP_PST_EXT.1, Storage of Persistent Information, requires the TSF to enumerate the minimum set of data that it must store persistently in order to function normally.

Management: FDP_PST_EXT.1

The following actions could be considered for the management functions in FMT:

- Enable and disable persistant storage of sensitive information.

Audit: FDP_PST_EXT.1

There are no auditable events foreseen.

FDP_PST_EXT.1 Storage of Persistent Information

Hierarchical to:	No other components.
Dependencies to:	No dependencies.

FDP_PST_EXT.1.1

The TSF shall provide the capability to operate without storing persistent data to the file system with the following exceptions: [**assignment**: *data that the TSF must store persistently*].

Appendix D - Entropy Documentation and Assessment

The TOE does not require any additional supplementary information to describe its entropy sources beyond the requirements outlined in the Base-PP.

Appendix E - Acronyms

Acronym	Meaning
Base-PP	Base Protection Profile
CC	Common Criteria
CEM	Common Evaluation Methodology
cPP	Collaborative Protection Profile
CRL	Certificate Revocation List
CSRF	Cross-Site Request Forgery
EP	Extended Package
FP	Functional Package
GPU	Graphics Processing Unit
HSTS	HTTP Strict Transport Security
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
IETF	Internet Engineering Task Force
IPC	Inter-Process Communication
OCSP	Online Certificate Status Protocol
OE	Operational Environment
PDF	Portable Document Format
PP	Protection Profile
PP-Configuration	Protection Profile Configuration
PP-Module	Protection Profile Module
SaaS	Software as a Service
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF Interface
TSS	TOE Summary Specification
W3C	World Wide Web Consortium
XSS	Cross-Site Scripting

Appendix F - Bibliography

Identifier	Title
[App PP]	Protection Profile for Application Software, Version 2.0, TBD
[CEM]	Common Methodology for Information Technology Security - Evaluation Methodology, CCMB-2022-11-006, CEM:2022, Revision 1, November 2022.
[CC]	Common Criteria for Information Technology Security Evaluation - • Part 1: Introduction and General Model, CCMB-2017-04-001, Version 3.1 Revision 5, April 2017. • Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1 Revision 5, April 2017. • Part 3: Security Assurance Components, CCMB-2017-04-003, Version 3.1 Revision 5, April 2017.