# PP-Module for Web Browsers



Version: 1.0
2021-06-18
**National Information Assurance Partnership**

# Revision History

| Version | Date | Comment |
|---------|------|---------|
| 1.0 | 2021-06-18 | Initial release as PP-Module |

# Contents

# 1 Introduction

## 1.1 Overview

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.

## 1.2 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. |
| 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. |
| Distributed TOE | A TOE composed of multiple components operating as a logical whole. |
| 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 Protection Profiles. |
| Security Assurance Requirement (SAR) | A requirement to assure the security of the TOE. |
| Security Functional Requirement | A requirement for security enforcement by the TOE. |

| (SFR) | |
|---|---|
| Security Target (ST) | A set of implementation-dependent security requirements for a specific product. |
| 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. |
| Target of Evaluation (TOE) | The product under evaluation. |

### 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 web pages 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 any web browser client capable of running on any operating system or platform and rendering web content using HTTP and HTTPS.

This PP-Module describes the extended security functionality of web browsers in terms of [CC]. As an extension of the App PP, it is expected that the content of this PP-Module will be appropriately combined with the App PP to include selection-based requirements in accordance with the selections and assignments made, and any optional and objective components to include: FCS_CKM.1.1, FCS_CKM.2.1, FCS_COP.1.1(*), FCS_DTLS_EXT.1.*, FCS_HTTPS_EXT.1.*, FCS_RBG_EXT.2.*, FCS_TLSC_EXT.1.*, FIA_X509_EXT.1.*, FIA_X509_EXT.2.*.

An ST must identify the applicable version of the App PP and this PP-Module in its conformance claims.

## 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**

This PP-Module inherits exact conformance as required from the specified Base-PP and as defined in the CC and CEM addenda for Exact Conformance, Selection-Based SFRs, and Optional SFRs (dated May 2017).

No additional PPs or PP-Modules are allowed to be specified in a PP-Configuration with this PP-Module aside from the Base-PP.

**CC Conformance Claims**

This PP-Module is conformant to Parts 2 (extended) and 3 (extended) of Common Criteria Version 3.1, Release 5 [CC].

**Package Claims**

This PP-Module is TLS Pacakge conformant.

# 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 Extended Package 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 even 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 in order 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/tabs or a faulty cross domain communications model can lead to leakage of content from one domain in one window/tab to a different domain in a different window/tab. Such attacks leverage the ability of browsers to display content from multiple domains simultaneously.

## 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.INTEGRITY**
    QQQQ

**O.MANAGEMENT**
    QQQQ

**O.PROTECTED_STORAGE**
    QQQQ

**O.PROTECTED_COMMS**
    QQQQ

**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 Operational Environment.

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

## 4.3 Security Objectives Rationale

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

**Table 1: Security Objectives Rationale**

| Threat, Assumption, or OSP | Security Objectives | Rationale |
| --- | --- | --- |
| T.FLAWED_ADDON | O.AAAAA | aaaaa |
| T.SAME_ORIGIN_VIOLATION | O.AAAAA | aaaaa |

# 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 PP Security Functional Requirements Direction

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

### 5.1.1 Modified SFRs

This PP-Module does not modify any SFRs defined by the PP.

## 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/tab;
- Local storage shall only be accessible from windows/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 web page 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**: *[**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 web page to access data in a second web page 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/disable storage of third party cookies | O | M |
| 2 | Enable/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/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/disable deletion of stored browsing data (cache, web form information) | O | M |
| 6 | Enable/disable storage of sensitive information (e.g., auto-fill, auto-complete) in persistent storage | O | O |
| 7 | Configure size of cookie cache | O | O |
| 8 | Configure size of cache | O | O |
| 9 | Enable/disable interaction with Graphic | O | O |

| | | | |
|---|---|---|---|
| | Processing Units (GPUs) | | |
| 10 | Configure the ability to advance to a website with an invalid or unvalidated X.509 certificate | O | O |
| 11 | Enable/disable establishment of a trusted channel if the browser cannot establish a connection to determine the validity of a certificate | O | O |
| 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/disable automatic installation of software updates and patches | O | O |
| 15 | Enable/disable ability for websites to register protocol handlers | O | O |
| 16 | Enable/disable display notification when unsigned, untrusted or unverified mobile code is encountered | O | O |
| 17 | Enable/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/disable launching of downloaded files outside the browser | O | O |
| 19 | Enable/disable JavaScript | O | O |
| 20 | Enable/disable [**selection**: *ActiveX, Flash, Java, [**assignment**: other mobile code types supported by the browser]*] mobile code | O | O |
| 21 | Enable/disable support for add-ons | O | O |
| 22 | Enable/disable individual add-ons | O | O |
| 23 | Enable/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 be 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**: *trusted add-ons*, *no add-ons*].

**Application Note:** FPT_AON_EXT.2 depends upon the selection made here. 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. Cryptographic verification (i.e., trust) of add-ons is tested in FPT_AON_EXT.2.1.

#### 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 to either save the file to the file system or not 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 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 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.INTEGRITY | FPT_DNL_EXT.1, FPT_MCD_EXT.1 | QQQQ |
| O.MANAGEMENT | FDP_TRK_EXT.1, FMT_MOF_EXT.1 | QQQQ |
| O.PROTECTED_STORAGE | FDP_COO_EXT.1, FDP_PST_EXT.1 | QQQQ |
| O.PROTECTED_COMMS | FCS_STS_EXT.1, FDP_STR_EXT.1, FPT_INT_EXT.1, FPT_INT_EXT.2 | QQQQ |
| O.DOMAIN_ISOLATION | FDP_ACF_EXT.1, FDP_SBX_EXT.1, FDP_SOP_EXT.1 | QQQQ |
| O.ADDON_INTEGRITY | FPT_AON_EXT.1, FPT_AON_EXT.2 | QQQQ |

# 6 Consistency Rationale

## 6.1 Protection Profile for

### 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.

### 6.1.2 Consistency of Security Problem Definition

| PP-Module Threat, Assumption, OSP | Consistency Rationale |
|---|---|
| T.FLAWED_ADDON | aaaaaa |
| T.SAME_ORIGIN_VIOLATION | aaaaaa |

### 6.1.3 Consistency of Objectives

The objectives for the TOEs are consistent with the PP based on the following rationale:

| PP-Module TOE Objective | Consistency Rationale |
|---|---|
| O.INTEGRITY | QQQQ |
| O.MANAGEMENT | QQQQ |
| O.PROTECTED_STORAGE | QQQQ |
| O.PROTECTED_COMMS | QQQQ |
| O.DOMAIN_ISOLATION | QQQQ |
| O.ADDON_INTEGRITY | QQQQ |

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 PP that are needed to support Web Browsers functionality. This is considered to be consistent because the functionality provided by the PP is being used for its intended purpose. The rationale for why this does not conflict with the claims defined by the PP are as follows:

| PP-Module Requirement | Consistency Rationale |
|---|---|
| **Modified SFRs** | |
| This PP-Module does not modify any requirements when the PP is the base. | |
| **Mandatory SFRs** | |
| FDP_ACF_EXT.1 | |
| FDP_COO_EXT.1 | |
| FDP_SBX_EXT.1 | |
| FDP_SOP_EXT.1 | |
| FDP_STR_EXT.1 | |
| FDP_TRK_EXT.1 | |
| FMT_MOF_EXT.1 | |
| FPT_AON_EXT.1 | |
| FPT_DNL_EXT.1 | |
| FPT_MCD_EXT.1 | |
| **Optional SFRs** | |
| FDP_PST_EXT.1 | |
| **Selection-based SFRs** | |
| FPT_AON_EXT.2 | |

### Objective SFRs

FCS_STS_EXT.1

FPT_INT_EXT.1

FPT_INT_EXT.2

### Implementation-Dependent SFRs

This PP-Module does not define any Implementation-Dependent requirements.

FCS_STS_EXT.1

FPT_INT_EXT.1

FPT_INT_EXT.2

### Implementation-Dependent SFRs

This PP-Module does not define any Implementation-Dependent requirements.

# 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). If a browser receives the HSTS header from a website, all future HTTP sessions between the browser and the domain or superdomain of that website must occur over TLS 1.2 (RFC 5246) or greater by using HTTPS (RFC 2818) negotiating the strongest cipher possible.

### 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 of 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; others may download updated lists of bad URLs at browser startup, updating the list periodically from the URL reputation service(s) 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 a selection in .*

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**: *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.

# Appendix C - Acronyms

| Acronym | Meaning |
| --- | --- |
| Base-PP | Base Protection Profile |
| CC | Common Criteria |
| CEM | Common Evaluation Methodology |
| CRL | Certificate Revocation List |
| CSRF | Cross-Site Request Forgery |
| 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 |
| SAR | Security Assurance Requirement |
| SFR | Security Functional Requirement |
| ST | Security Target |
| SaaS | Software as a Service |
| 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 D - Bibliography

| Identifier | Title |
|---|---|
| [App PP] | Protection Profile for Application Software, Version 1.3, March 1, 2019 |
| [CEM] | Common Methodogy for Information Technology Security Evaluation, Version 3.1r5, CCMB-2017-04-004, April 2017 |
| [CC] | Common Criteria for Information Technology Security Evaluation -<br>• Part 1: Introduction and General Model, CCMB-2017-04-001, Version 3.1 Revision 5, April 2017.<br>• Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1 Revision 5, April 2017.<br>• Part 3: Security Assurance Components, CCMB-2017-04-003, Version 3.1 Revision 5, April 2017. |