



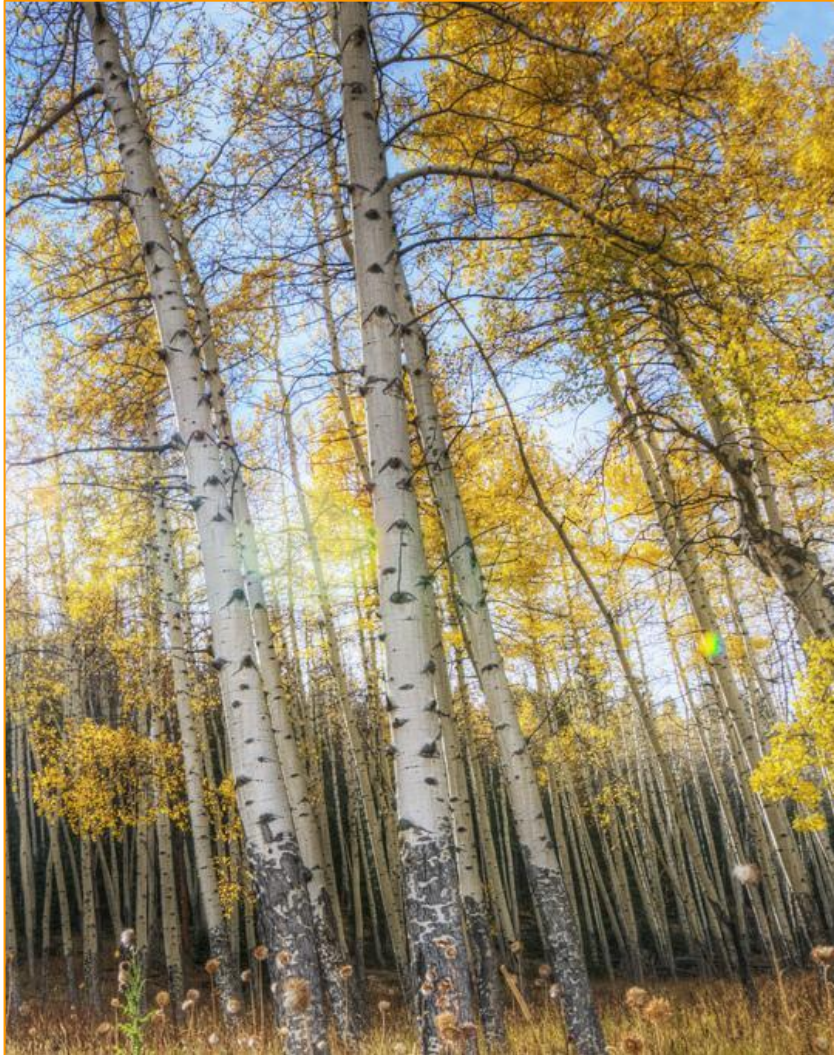
從 SSDLC 視角 簡介 FIPS 140-3 標準



亞洲最佳資安公司金獎

Realize Ultimate Security every step starts with the labs

/usr/bin/whoami



Aspen Yang

aspen [dot] yang [at] onwardsecurity [dot] com
Core Technology Div.

系統軟體研發工程師、後端程式研發工程師、
資安產品 Test Case 研發與 FAE 工程師、
SRE、DevOps、MIS、
資安研究員、資安顧問

Skill sets:

Java (Backend | Spring Boot)、
C# (System | .Net Framework) 、
Python、SQL、Bash

Photo by Yuya Sekiguchi (Creative Commons Attribution 2.0 Generic)

CONTENT

1

Prologue

2

NIST SP 800-218: SSDF v1.1

3

Endpoint Hardening:
FIPS Mode & FIPS-Compliant

4

Introduction of NIST FIPS 140-3

5

Summary

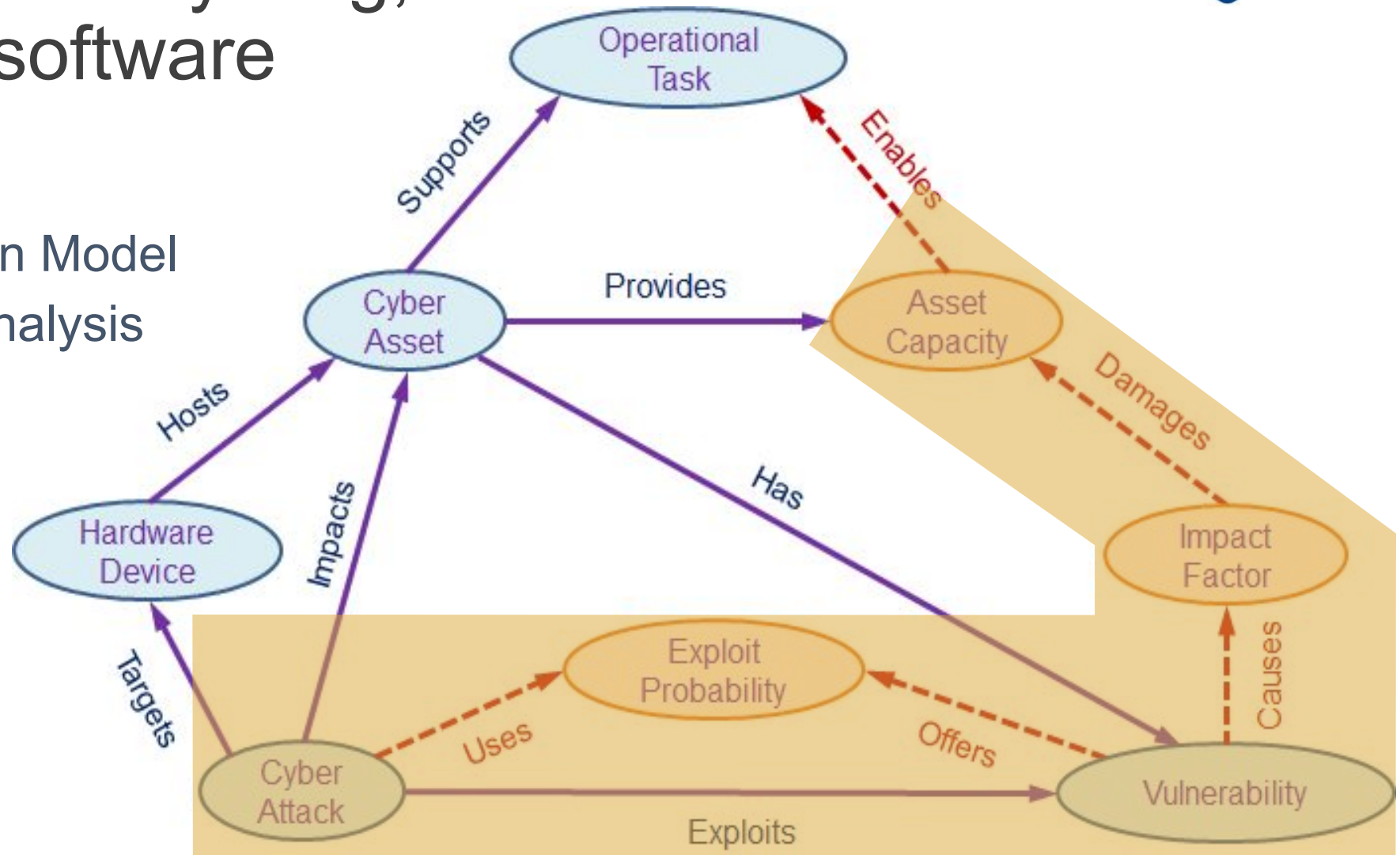
1

Prologue

Software drive everything, Risk in every software

《Metrics of Security》

Attack Risk Prediction Model
for Mission Impact Analysis

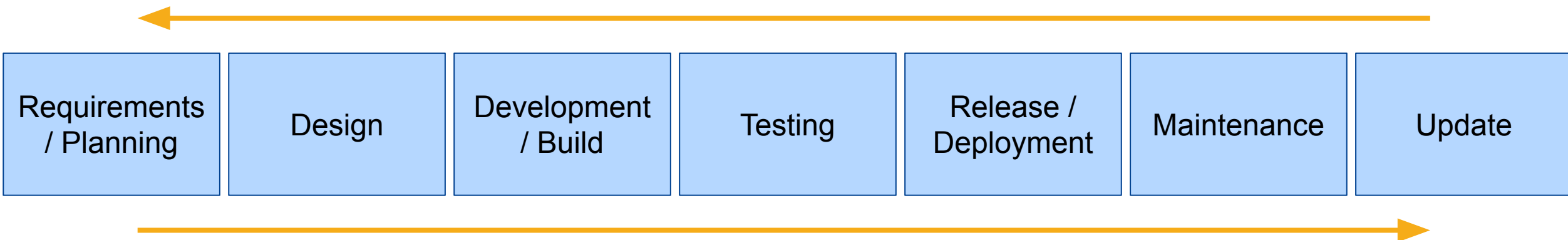


Cheng, Y. , Deng, J. , Li, J. , DeLoach, S. , Singhal, A. and Ou, X. (2014), Metrics of Security, Cyber Defense and Situational Awareness, Springer, Dusseldorf, -1, [online], https://doi.org/10.1007/978-3-319-11391-3_13, https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=917850

Main Goal of Secure SDLC

- 減少弱點的產生(Minimize Vulnerabilities)
- 保護軟體智慧財產(Protect Software Intellectual Property)
- 降低因弱點導致的內部與外部成本：早期發現、早期治療
- 尋求「階段步驟(Stages)、方法論(Methodologies)、最佳實踐(Best Practices)」
以保護資料：在資料的三態(儲存、使用、傳輸)增加安全性處理

The early security issue that is addressed, the less effort and cost is required.



Software Development Life Cycle

Examples of a Secure SDLC

- NIST Secure Software Development Framework [**SSDF**]
<https://csrc.nist.gov/publications/detail/sp/800-218/final>
- Microsoft Security Development Lifecycle [**MS SDL**]
<https://www.microsoft.com/en-us/securityengineering/sdl/practices>
- OWASP Software Assurance Maturity Model [**OWASP SAMM**]
<https://owasp.org/www-project-samm/>
- Payment Card Industry (PCI) Security Standards Council (2021)
Secure Software Lifecycle (Secure SLC) Requirements and Assessment Procedures
[**PCI SSLC**]
https://www.pcisecuritystandards.org/document_library?category=sware_sec#results
- IEC 62443-4-1:2018 “Secure Product Development Lifecycle Requirements”
<https://webstore.iec.ch/publication/33615>
-

2

NIST SP 800-218: SSDF v1.1

NIST & NIST Information Technology Laboratory



制定美國政府與科技之標準的研究院

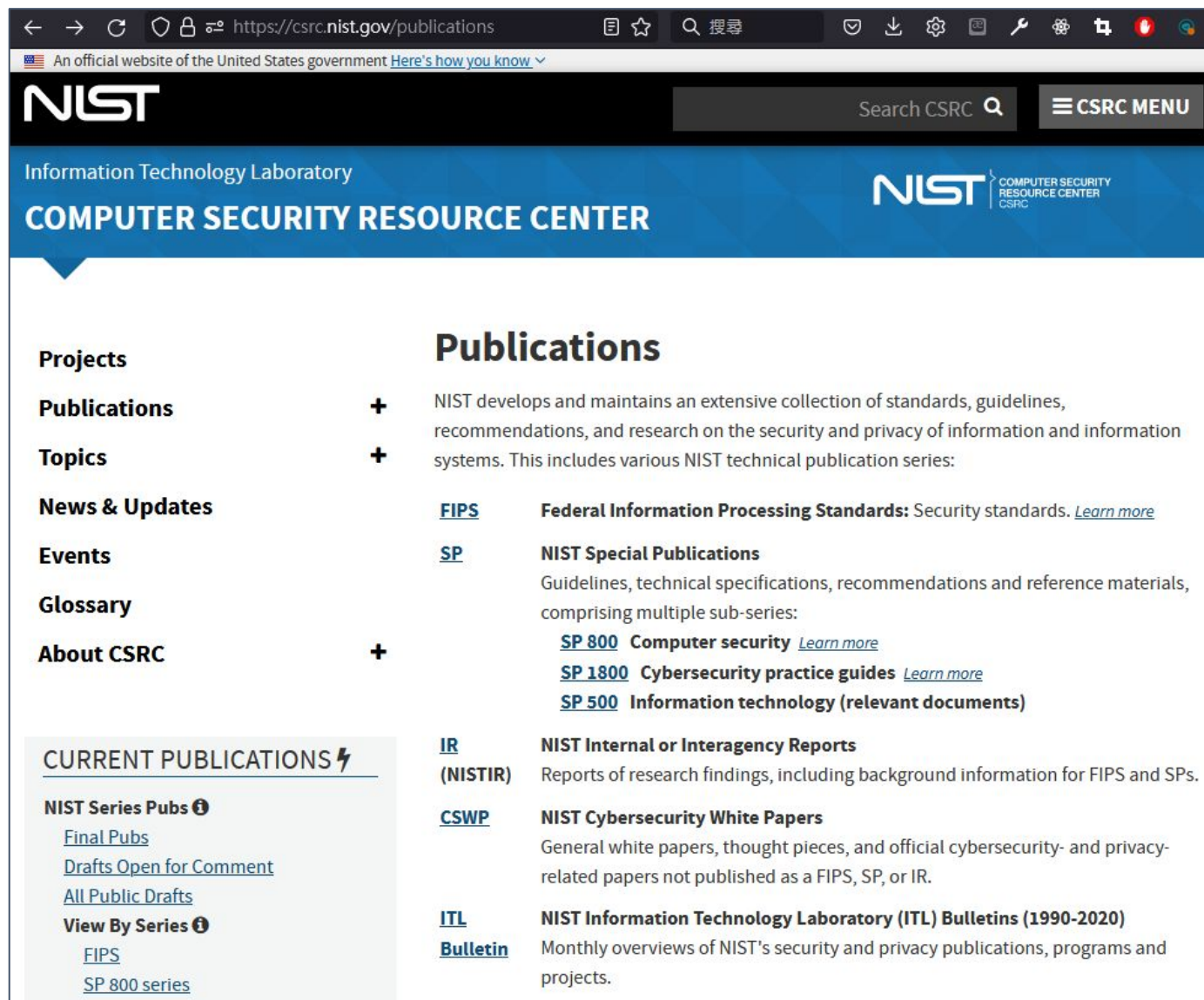
FIPS: 聯邦資訊處理標準

SP: NIST 特殊出版品

IR: NIST 內部報告或跨獨立機關報告

CSWP: NIST 資安白皮書(草案)

ITL Bulletin: ITL 每月資安要點公告
(短文, 已停刊)



The screenshot shows the NIST Computer Security Resource Center (CSRC) website. The browser address bar displays 'https://csrc.nist.gov/publications'. The page header includes the NIST logo, a search bar labeled 'Search CSRC', and a 'CSRC MENU' button. The main navigation bar identifies the 'Information Technology Laboratory' and 'COMPUTER SECURITY RESOURCE CENTER'. The page content is organized into two columns. The left column lists navigation options: Projects, Publications, Topics, News & Updates, Events, Glossary, and About CSRC. The right column features a 'Publications' section with a descriptive paragraph and a list of publication series: FIPS (Federal Information Processing Standards), SP (NIST Special Publications), IR (NIST Internal or Interagency Reports), CSWP (NIST Cybersecurity White Papers), and ITL Bulletin (NIST Information Technology Laboratory Bulletins). A 'CURRENT PUBLICATIONS' section is also visible at the bottom left, listing 'Final Pubs', 'Drafts Open for Comment', and 'All Public Drafts'.

NIST-ITL DevSecOps Publications

包括

- 知名的 SP 800-207 零信任架構
- SP 800-218 SSDF的白皮書
- SP 800-160系列:系統安全工程
- 許多雲端資安的軟硬體標準與指南

DevSecOps



Publications

The following NIST-authored publications are directly related to this project.

Series & Number	Title	Status	Released
SP 800-207	Zero Trust Architecture	Final	08/11/2020
SP 800-204A	Building Secure Microservices-based Applications Using Service-Mesh Architecture	Final	05/27/2020
White Paper NIST CSWP 14 ipd (Draft)	Hardware-Enabled Security for Server Platforms: Enabling a Layered Approach to Platform Security for Cloud and Edge Computing Use Cases	Draft	04/28/2020
White Paper NIST CSWP 13	Mitigating the Risk of Software Vulnerabilities by Adopting a Secure Software Development Framework (SSDF)	Withdrawn	04/23/2020
SP 800-160 Vol. 2	Developing Cyber Resilient Systems: A Systems Security Engineering Approach	Withdrawn	11/27/2019
SP 800-204	Security Strategies for Microservices-based Application Systems	Final	08/07/2019
SP 800-125A Rev. 1	Security Recommendations for Server-based Hypervisor Platforms	Final	06/07/2018
SP 800-160 Vol. 1	Systems Security Engineering: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems	Final	03/21/2018
SP 800-190	Application Container Security Guide	Final	09/25/2017
SP 800-125B	Secure Virtual Network Configuration for Virtual Machine (VM) Protection	Final	03/07/2016
SP 800-40 Rev. 3	Guide to Enterprise Patch Management Technologies	Withdrawn	07/22/2013
SP 800-125	Guide to Security for Full Virtualization Technologies	Final	01/28/2011

NIST Secure Software Development Framework

- 自從白皮書提出，歷時近兩年，今年2022年二月，審定v1.1最終版
- 支援多樣SSDLC標準因而具有延展性，可依照參照的SSDLC標準進行文件化，適用範圍可大可小
- 不指定程式語言、軟體開發流程框架、實作軟體、開發環境與維運環境，可以整合現有工具鏈與工作流程
- 適用資訊科技(IT)、工控系統(ICS)、網路-實體整合系統(Cyber-Physical Systems, CPS)、物聯網(IoT)



An official website of the United States government [Here's how you know](#) ▾

NIST Search CSRC 🔍 CSRC MENU

Information Technology Laboratory
COMPUTER SECURITY RESOURCE CENTER

PUBLICATIONS

SP 800-218

Secure Software Development Framework (SSDF) Version 1.1: Recommendations for Mitigating the Risk of Software Vulnerabilities

f t

Date Published: February 2022

Supersedes: [White Paper NIST CSWP 13 \(04/23/2020\)](#)

Author(s)
Murugiah Souppaya (NIST), Karen Scarfone (Scarfone Cybersecurity), Donna Dodson

Abstract
Few software development life cycle (SDLC) models explicitly address software security in detail, so secure software development practices usually need to be added to each SDLC model to ensure that the software being developed is well-secured. This document recommends the Secure Software Development Framework (SSDF) – a core set of high-level secure software development practices that can be integrated into each SDLC implementation. Following these practices should help software producers reduce the number of vulnerabilities in released software, mitigate the potential

DOCUMENTATION

Publication:
[SP 800-218 \(DOI\)](#)
[Local Download](#)

Supplemental Material:
[Potential updates \(xls\)](#)
[SP 800-218 Table in Excel \(xls\)](#)
[Delta from April 2020 paper \(word\)](#)
[Delta from September 2021 public draft \(word\)](#)
[SSDF Project homepage \(other\)](#)
[Executive Order 14028, Improving the Nation's Cybersecurity \(web\)](#)

NIST Secure Software Development Framework

將實踐(Practices)進行四個分群：

- 組織要作的準備 (PO) x 5
- 保護軟體 (PS) x 3
- 產出良好安全的軟體 (PW) x 9
- 回應弱點 (RV) x 3

每個實踐以這些元素組成：

- 實踐 (Practice)
- 工作 (Tasks)
- 概念實施範例
(Notional Implement Examples)
- 參照 (References)

2 The Secure Software Development Framework

This document defines version 1.1 of the Secure Software Development Framework (SSDF) with fundamental, sound, and secure recommended practices based on established secure software development practice documents. The practices are organized into four groups:

1. **Prepare the Organization (PO):** Organizations should ensure that their people, processes, and technology are prepared to perform secure software development at the organization level. Many organizations will find some PO practices to also be applicable to subsets of their software development, like individual development groups or projects.
2. **Protect the Software (PS):** Organizations should protect all components of their software from tampering and unauthorized access.
3. **Produce Well-Secured Software (PW):** Organizations should produce well-secured software with minimal security vulnerabilities in its releases.
4. **Respond to Vulnerabilities (RV):** Organizations should identify residual vulnerabilities in their software releases and respond appropriately to address those vulnerabilities and prevent similar ones from occurring in the future.

Each practice definition includes the following elements:

- **Practice:** The name of the practice and a unique identifier, followed by a brief explanation of what the practice is and why it is beneficial
- **Tasks:** One or more actions that may be needed to perform a practice
- **Notional Implementation Examples:** One or more notional examples of types of tools, processes, or other methods that could be used to help implement a task. No examples or combination of examples are required, and the stated examples are not the only feasible options. Some examples may not be applicable to certain organizations and situations.
- **References:** Pointers to one or more established secure development practice documents and their mappings to a particular task. Not all references will apply to all instances of software development.

NIST Secure Software Development Framework

Implementation Examples Tasks Practice

- Design Software to Meet Security Requirements and Mitigate Security Risks (PW.1)
- Review the Software Design to Verify Compliance with Security Requirements and Risk Information (PW.2)
- Verify Third-Party Software Complies with Security Requirements (PW.3)
- Reuse Existing, Well-Secured Software When Feasible Instead of Duplicating Functionality (PW.4)
- Create Source Code by Adhering to Secure Coding Practices (PW.5)
- Configure the Compilation, Interpreter, and Build Processes to Improve Executable Security (PW.6)
- Review and/or Analyze Human-Readable Code to Identify Vulnerabilities and Verify Compliance with Security Requirements (PW.7)
- Test Executable Code to Identify Vulnerabilities and Verify Compliance with Security Requirements (PW.8)
- Configure Software to Have Secure Settings by Default (PW.9)

Produce Well-Secured Software (PW)



Prepare the Organization (PO)

Practice Tasks Notional Implementation Examples References

- Define Security Requirements for Software Development (PO.1)
- Implement Roles and Responsibilities (PO.2)
- Implement Supporting Toolchains (PO.3)
- Define and Use Criteria for Software Security Checks (PO.4)
- Implement and Maintain Secure Environments for Software Development (PO.5)

- PO.5.1: Separate and protect each environment involved in software development.
- PO.5.2: Secure and harden development endpoints (i.e., endpoints for software designers, developers, testers, builders, etc.) to perform development-related tasks using a risk-based approach.

Protect the Software (PS)

Practice Tasks Notional Implementation Examples References

- Protect All Forms of Code from Unauthorized Access and Tampering (PS.1)
- Provide a Mechanism for Verifying Software Release Integrity (PS.2)
- Archive and Protect Each Software Release (PS.3)

Implementation Examples Tasks Practice

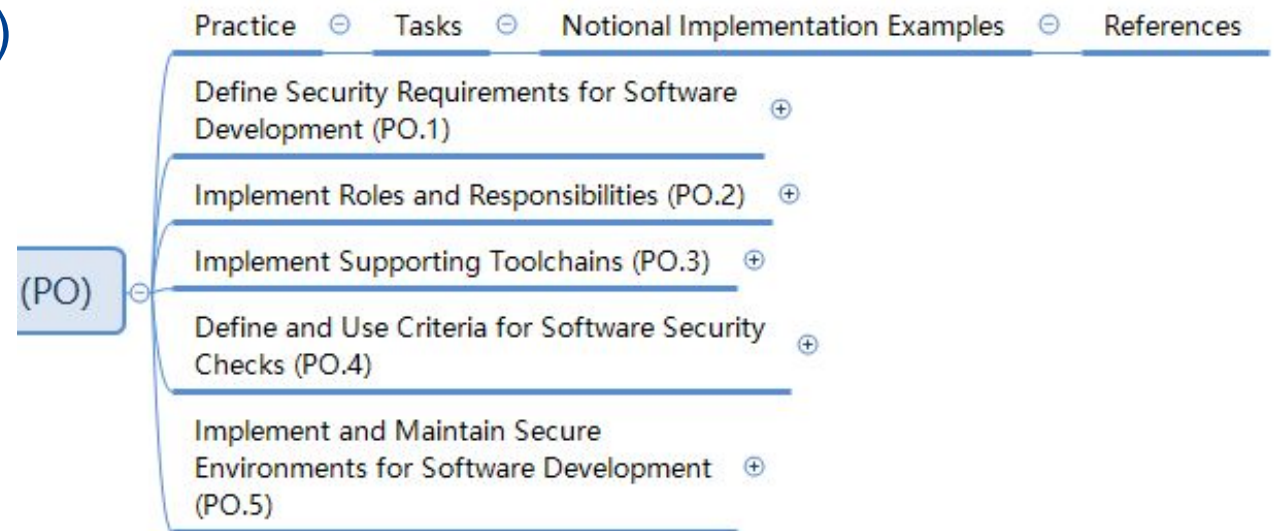
- Identify and Confirm Vulnerabilities on an Ongoing Basis (RV.1)
 - Assess, Prioritize, and Remediate Vulnerabilities (RV.2)
- Analyze Vulnerabilities to Identify Their Root Causes (RV.3)

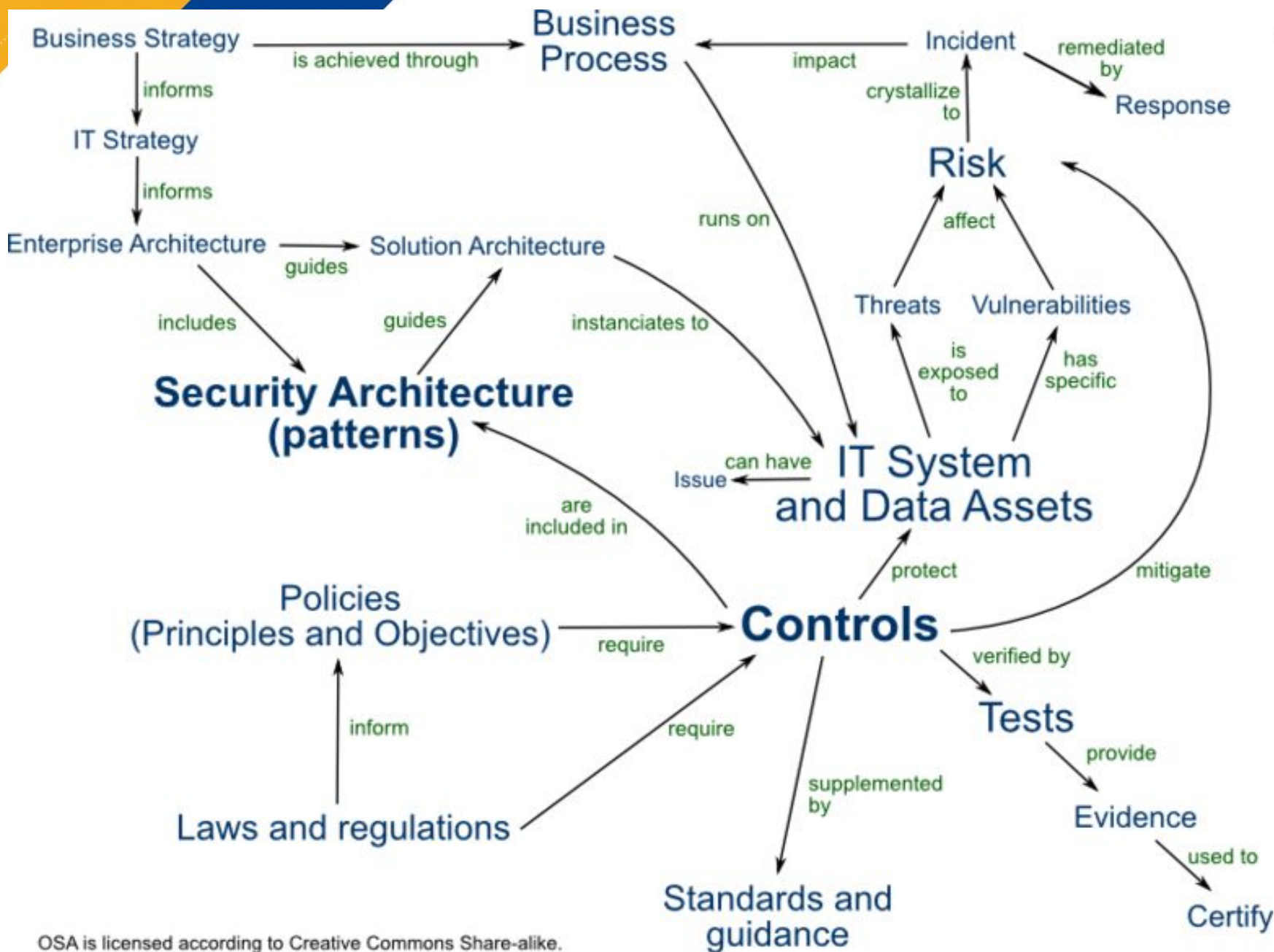
Respond to Vulnerabilities (RV)

組織要作的準備 (PO)

實踐(Practices)

- 定義軟體開發的安全需求 (PO.1)
- 實施角色與職責 (PO.2)
- 實施支援工具鏈 (PO.3)
- 定義與使用軟體的安全標準查核表 (PO.4)
- 實施與維護軟體開發的安全環境 (PO.5)





SSDLC基本資安需求

- 安全架構與安全設計
- 資安控制措施
- 軟體弱點辨識
- 以不可逆方式保存密碼
- 進行身分認證
- 權限設計與授權管理
- 用金鑰加解密
- 機密敏感性檔案處理
- 安全連接與通訊
- 安全傳輸檔案
- 簽署資料
- 驗證資料完整性
- 數位證據保存

OSA is licensed according to Creative Commons Share-alike.

Please see: <http://www.opensecurityarchitecture.org/cms/about/license-terms>.

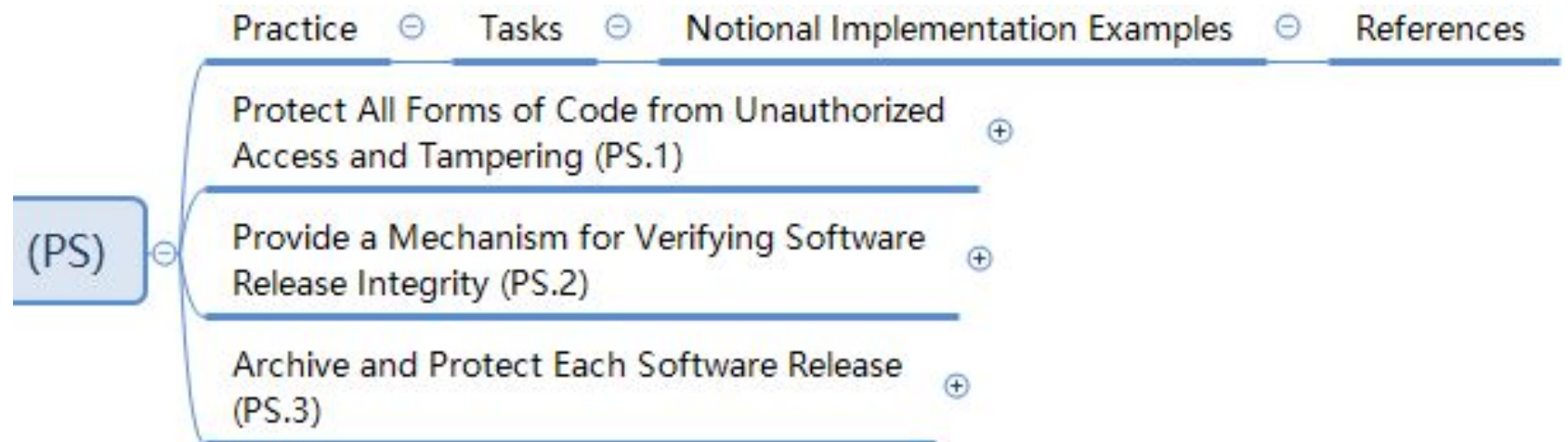
© 2022 Onward Security Corp. Creative Commons CC0

<https://www.opensecurityarchitecture.org/cms/foundations/osa-taxonomy>

保護軟體 (PS)

實踐(Practices)

- 保護所有狀態的程式碼以防範未經授權的存取與竄改 (PS.1)
- 提供機制驗證軟體釋出的完整性 (PS.2)
- 每次的軟體釋出都進行庫存與保護 (PS.3)



Software Bill Of Materials

認識「軟體物料清單」(SBOM)

- 可追蹤軟體與函式庫的相依性
- ISO/IEC 5962:2021, Linux Foundation, SPDX - Software Package Data Exchange
- NIST, SWID - Software Identification Tags
- OWASP Cyclone DX

弱點可利用性資訊交換標準 (Vulnerability Exploitability eXchange, VEX)

- 將SBOM與弱點資料庫進行關聯
- 讓SBOM變得有用, 使得供應鏈資安透明化



The screenshot shows the CISA website page for Software Bill of Materials. The page features the CISA logo and the text "CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY". The main heading is "SOFTWARE BILL OF MATERIALS". The content explains that SBOM is a key building block in software security and supply chain risk management, and that CISA will advance this work through community engagement and development. It also mentions the Vulnerability Exploitability eXchange (VEX) and provides contact information for SBOM@cisadhs.gov.

Cybersecurity > Software Bill of Materials

Cybersecurity

- [Cybersecurity Training & Exercises](#)
- [Cybersecurity Summit 2020](#)
- [Cyber QSMO Marketplace](#)
- [Combating Cyber Crime](#)
- [Securing Federal Networks](#)
- [Protecting Critical Infrastructure](#)

SOFTWARE BILL OF MATERIALS

A “software bill of materials” (SBOM) has emerged as a key building block in software security and software supply chain risk management. A SBOM is a nested inventory, a list of ingredients that make up software components. The SBOM work has advanced since 2018 as a collaborative community effort, driven by [National Telecommunications and Information Administration’s \(NTIA\) multistakeholder process](#).

CISA will advance the SBOM work by facilitating community engagement, development, and progress, with a focus on scaling and operationalization, as well as tools, new technologies, and new use cases. This website will also be a nexus for the broader set of SBOM resources across the digital ecosystem and around the world.

An SBOM-related concept is the [Vulnerability Exploitability eXchange \(VEX\)](#). A VEX document is an attestation, a form of a security advisory that indicates whether a product or products are affected by a known vulnerability or vulnerabilities. For more information on how to receive updates or join in on the efforts around VEX, please contact SBOM@cisadhs.gov.

產出良好安全的軟體 (PW)

實踐(Practices)

- 軟體設計:與資安需求相符並緩解資安風險 (PW.1)
- 複驗軟體設計:驗證合規於資安需求與風險資訊 (PW.2)
- 驗證第三方軟體於滿足資安需求的條件下進行編譯 (PW.3)
- 以可行的功能替代複製功能, 重複使用已存在且安全性良好的軟體 (PW.4)
- 寫源碼的時候要堅持程式設計資安實踐 (PW.5)
- 直譯、編譯和組建的流程, 設定為可提高執行期資安的組態 (PW.6)
- 複驗和分析人類可讀的程式碼以辨認弱點, 並驗證合規於資安需求 (PW.7)
- 測試可執行的程式以辨認弱點, 並驗證合規於資安需求 (PW.8)
- 軟體的預設值設定必須是安全組態 (PW.9)

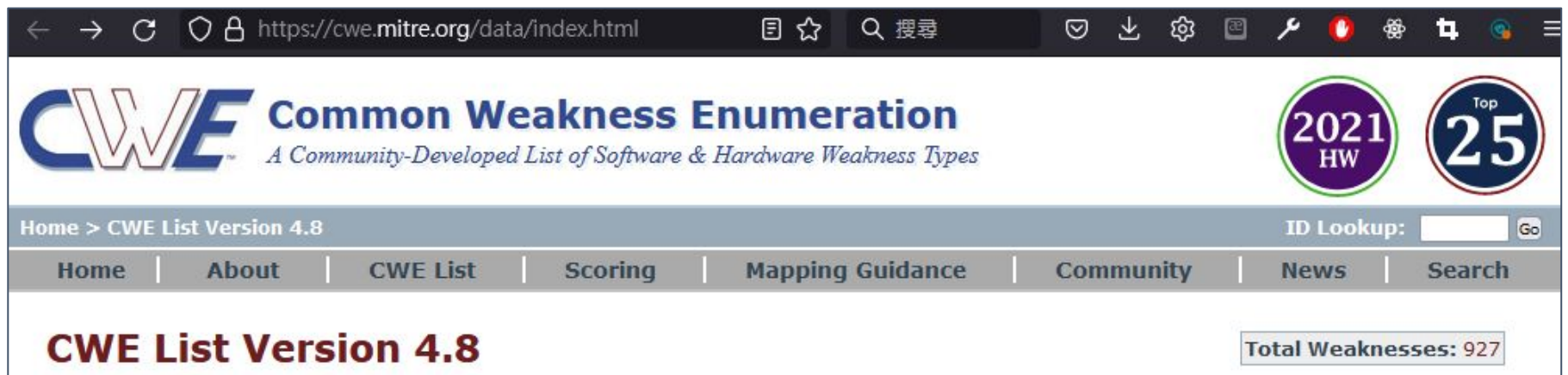
Common Weakness Enumeration

認識「通用缺陷列表」(CWE)

- 軟、硬體錯誤之缺陷(Weakness; 根因) 導致弱點(Vulnerability; 漏洞)發生

後設來說, 實務上會產生軟、硬體缺陷的主要原因:

- 隕石式開發流程 (特急!)
- 未導入安全開發流程
- 開發與維運人員缺乏資安領域知識
- 人力資源不足
- 一時粗心
-

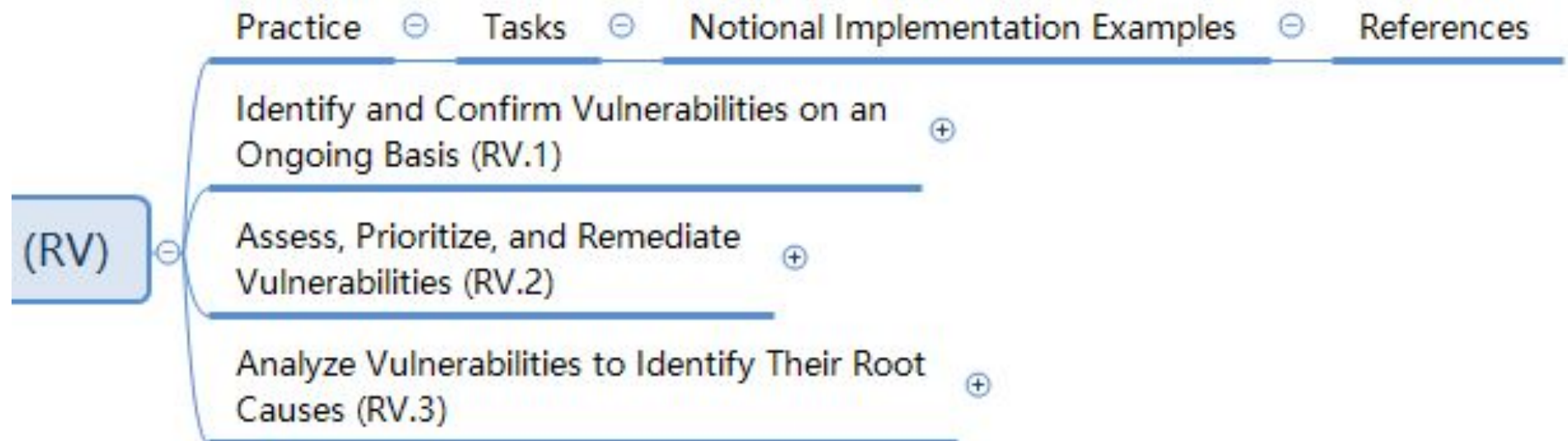


The screenshot shows the homepage of the Common Weakness Enumeration (CWE) website. The URL in the browser is <https://cwe.mitre.org/data/index.html>. The page features the CWE logo and the text "Common Weakness Enumeration" and "A Community-Developed List of Software & Hardware Weakness Types". There are two circular badges: one for "2021 HW" and another for "Top 25". The navigation menu includes "Home", "About", "CWE List", "Scoring", "Mapping Guidance", "Community", "News", and "Search". The main content area displays "CWE List Version 4.8" and "Total Weaknesses: 927".

回應弱點 (RV)

實踐(Practices)

- 於現有的基礎持續辨認與確認弱點 (RV.1)
- 評估、訂定優先順序與修復弱點 (RV.2)
- 分析弱點以辨認根因 (RV.3)



Common Vulnerabilities and Exposures

認識「通用漏洞與披露」(CVE)

- 被發現的弱點進行編號
CVE-YYYY-NNNNN
- 進行弱點描述

NIST的國家弱點資料庫

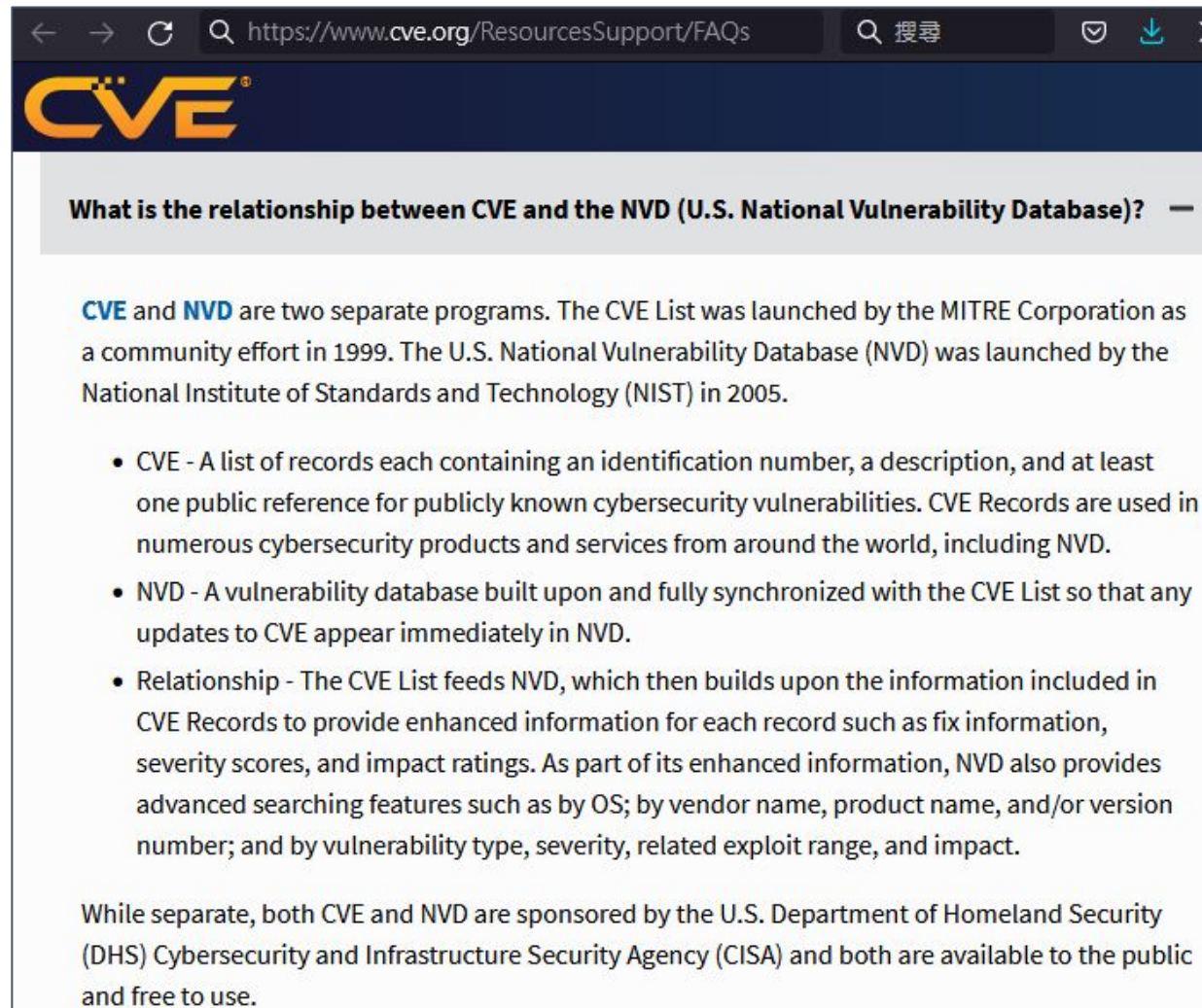
(National Vulnerabilities Database, NVD)

- 進行評分與根因列舉

通用弱點評分系統

(Common Vulnerability Scoring System, CVSS)

- 計算弱點影響的評分，目前為第三版



← → ↻ 🔍 <https://www.cve.org/ResourcesSupport/FAQs> 🔍 搜尋

What is the relationship between CVE and the NVD (U.S. National Vulnerability Database)?

CVE and **NVD** are two separate programs. The CVE List was launched by the MITRE Corporation as a community effort in 1999. The U.S. National Vulnerability Database (NVD) was launched by the National Institute of Standards and Technology (NIST) in 2005.

- **CVE** - A list of records each containing an identification number, a description, and at least one public reference for publicly known cybersecurity vulnerabilities. CVE Records are used in numerous cybersecurity products and services from around the world, including NVD.
- **NVD** - A vulnerability database built upon and fully synchronized with the CVE List so that any updates to CVE appear immediately in NVD.
- **Relationship** - The CVE List feeds NVD, which then builds upon the information included in CVE Records to provide enhanced information for each record such as fix information, severity scores, and impact ratings. As part of its enhanced information, NVD also provides advanced searching features such as by OS; by vendor name, product name, and/or version number; and by vulnerability type, severity, related exploit range, and impact.

While separate, both CVE and NVD are sponsored by the U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA) and both are available to the public and free to use.

3

Endpoint Hardening: FIPS Mode & FIPS-Compliant

Endpoints Hardening is Required

- NIST SSDF的PO 5.2要求「研發工作相關端點，基於風險的考量，進行安全強固化」(Secure and Harden); PO 5.2其中一個概念實施範例，建議使用FIPS合規的加密，保護所有處於儲存與傳輸狀態的敏感資料
- 國際組織SWIFT與國內金管會，皆要求銀行業使用認證規格FIPS 140-2 Level 3以上，已通過認證的「硬體安全模組」(HSM) 以儲存重要金鑰
- GCP雲端的「虛擬信賴平台模組」(vTPM) 通過FIPS 140-2 Level 1認證規格，使用vTPM可以有效防範雲端的虛擬機受到rootkit攻擊

Practices	Tasks	Notional Implementation Examples	References
Implement and Maintain Secure Environments for Software Development (PO.5): Ensure that all components of the environments for software development are strongly protected from internal and external threats to prevent compromises of the environments or the software being developed or maintained within them. Examples of environments for software development include development, build, test, and distribution environments.	PO.5.2: Secure and harden development endpoints (i.e., endpoints for software designers, developers, testers, builders, etc.) to perform development-related tasks using a risk-based approach.	from attempted and actual cyber incidents. Example 8: Configure security controls and other tools involved in separating and protecting the environments to generate artifacts for their activities. Example 9: Continuously monitor all software deployed in each environment for new vulnerabilities, and respond to vulnerabilities appropriately following a risk-based approach. Example 10: Configure and implement measures to secure the environments' hosting infrastructures following a zero trust architecture ⁷ .	
		Example 1: Configure each development endpoint based on approved hardening guides, checklists, etc.; for example, enable FIPS-compliant encryption of all sensitive data at rest and in transit. Example 2: Configure each development endpoint and the development resources to provide the least functionality needed by users and services and to enforce the principle of least privilege. Example 3: Continuously monitor the security posture of all development endpoints, including monitoring and auditing all use of privileged access. Example 4: Configure security controls and other tools involved in securing and	BSAFSS: DE-1-1, IA.1, IA.2 EO14028: 4e(i)(C), 4e(i)(E), 4e(i)(F), 4e(ii), 4e(iii), 4e(v), 4e(vi), 4e(ix) IEC62443: SM-7 NISTCSF: PR.AC-4, PR.AC-7, PR.IP-1, PR.IP-3, PR.IP-12, PR.PT-1, PR.PT-3, DE.CM SCAGILE: Tasks Requiring the Help of Security Experts 11 SCSIC: Vendor Software Delivery Integrity Controls SP80053: SA-15 SP800161: SA-15

About FIPS 140

- **FIPS:**
美國聯邦資訊處理標準(Federal Information Processing Standard)
- **FIPS 140 (140-2、140-3):**
密碼學模組之資安要求 (Security Requirements for Cryptographic Modules)
1994年FIPS 140發布、2001年FIPS 140-2發布、2019年FIPS 140-3發布
→ 密碼學模組(Cryptographic Modules)
可以是「硬體、韌體、軟體、軟體與硬體的混合、韌體與硬體的混合」
- **CMVP:**
NIST-ITL的「密碼學模組驗證程序」(Cryptographic Module Validation Program)
- **CAVP:**
NIST-ITL的「密碼學演算法驗證程序」(Cryptographic Algorithm Validation Program)

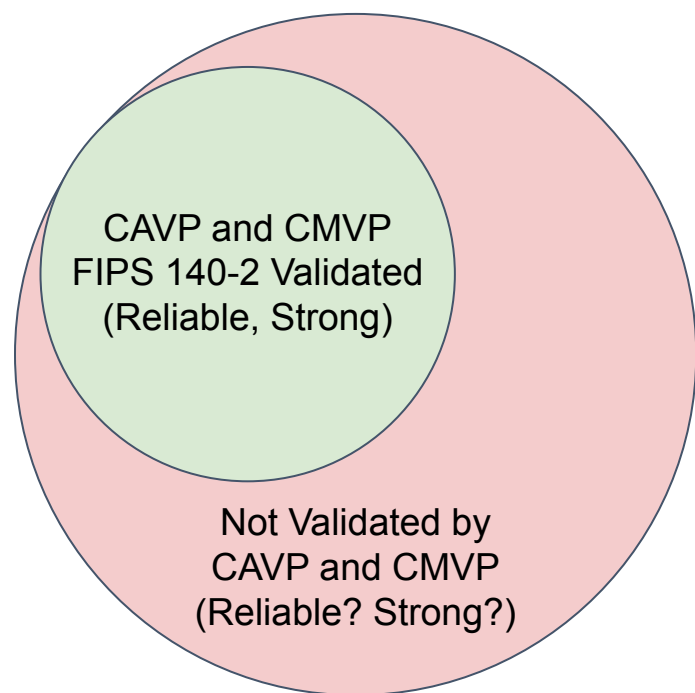
FIPS 140-2 Validated v.s. FIPS-Compliant

- 作業系統(OS)要在美國營利上市, 就有機率會保存「受美國政府管控由美國政府或代表單位生成的資料」, 即需要包含驗證過的密碼學模組(FIPS 140-2 Validated)
- 若某程式使用驗證過的密碼學模組, 則此程式稱為 FIPS-Compliant, 通譯為「**FIPS合規**」或「**FIPS相容**」
- 因此, OS有所謂的「FIPS 模式」、安全連線程式(SSH、VPN)有所謂的「FIPS 相容編譯版」(FIPS-Compliant compiled version)或是自帶CMVP驗證過的密碼學模組
- Firefox自帶CMVP驗證過的密碼學模組, 當OS啟用FIPS模式, 能設定Mozilla的Network Security Service (NSS)也啟用FIPS模式, 作為FIPS合規的資料傳輸; Chrome(Chromium/Edge)使用自帶CMVP驗證過的BoringSSL密碼學模組(Forked OpenSSL)



```
[aspen@OracleLinux ~]$ openssl version
OpenSSL 1.1.1k FIPS 25 Mar 2021
[aspen@OracleLinux ~]$ ssh -U
OpenSSH_8.0p1, OpenSSL 1.1.1k FIPS 25 Mar 2021
[aspen@OracleLinux ~]$ sudo sshd -dt
debug1: sshd version OpenSSH_8.0, OpenSSL 1.1.1k FIPS 25 Mar 2021
```

FIPS 140-2 Validated v.s. FIPS-Compliant

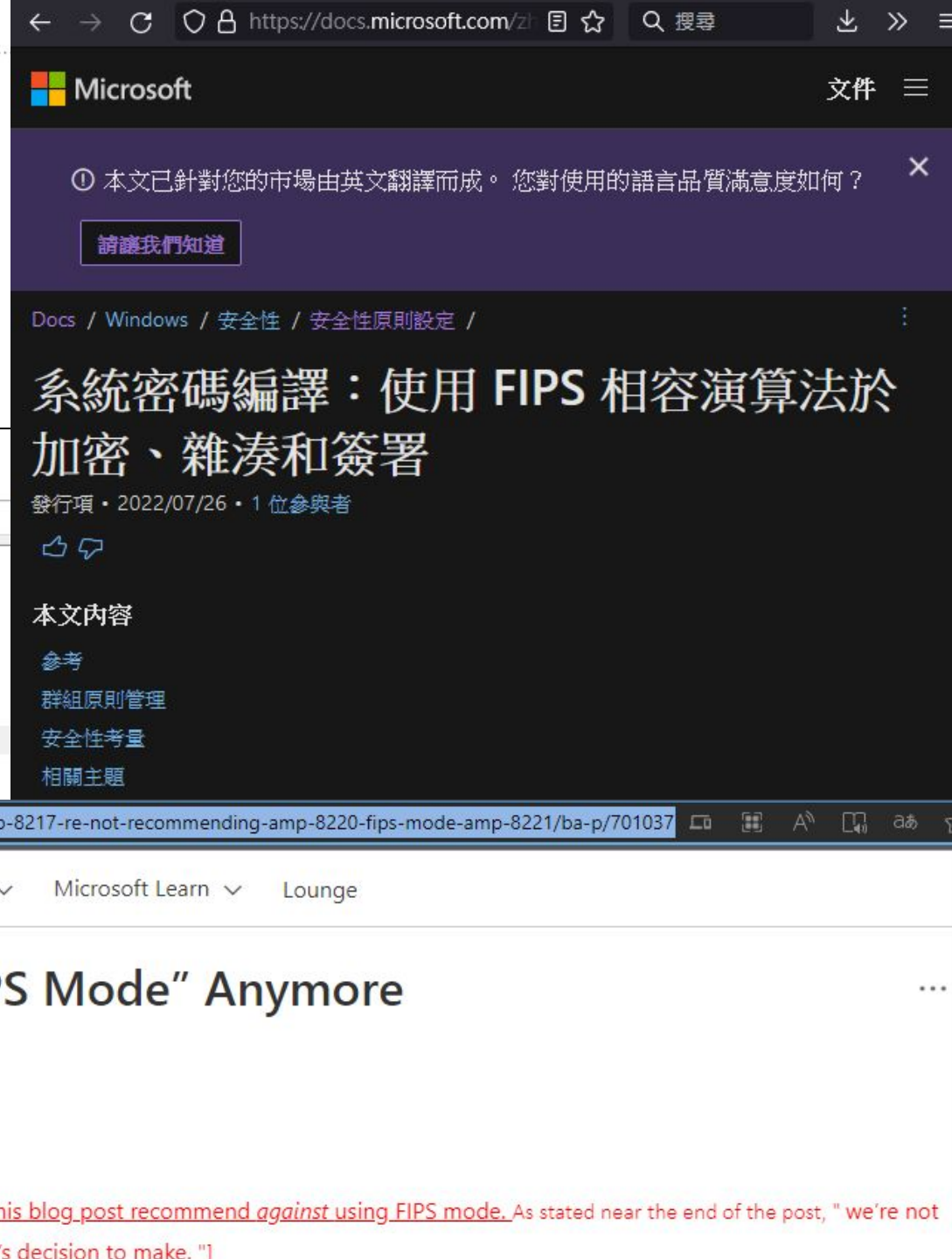
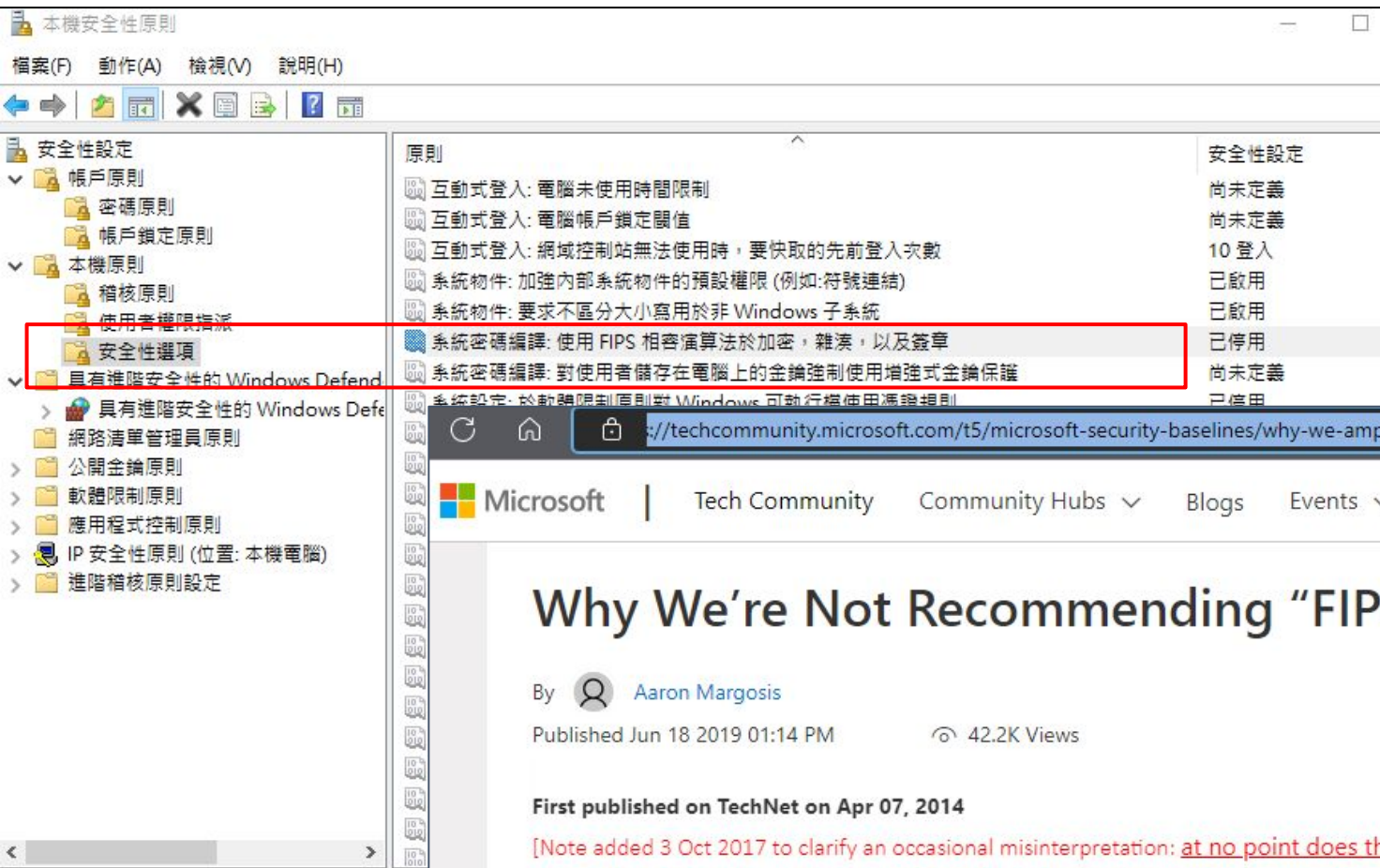


The Cryptography Algorithms
within
The Cryptography Modules
(Venn Diagram)

- 目前常見的OS都有FIPS模式能啟用，部分OS甚至預設啟用；在OS內「啟用FIPS模式」意謂「在OS內的程式，如果需要調用密碼學演算法，像是金鑰生成、加解密、單向雜湊，被要求**只能使用CMVP驗證過可信的**(Security Reliable)密碼學模組，在OS內的程式必須**透過此密碼學模組的資安政策**(Security Policy)與**密碼學模組介面**(Cryptographic module interfaces)，進行調用此密碼學模組內已實作之密碼學演算法。一般來說，啟用FIPS模式必須合規，**只允許在OS內的程式調用CAVP驗證過的高強度密碼學演算法，作為FIPS核可的操作模式**」
- 根據不同密碼學模組實作的狀況與使用情境，密碼學模組有可能仍然會提供在OS內的程式調用弱強度之密碼學演算法，通常是為了相容性，作為非FIPS核可的操作模式

FIPS Mode

Microsoft Windows預設不啟用，要自行啟用，啟用後不需要重開機



Why We're Not Recommending "FIPS Mode" Anymore

By  Aaron Margosis

Published Jun 18 2019 01:14 PM

42.2K Views

First published on TechNet on Apr 07, 2014

[Note added 3 Oct 2017 to clarify an occasional misinterpretation: at no point does this blog post recommend against using FIPS mode. As stated near the end of the post, "we're not telling customers to turn it off – our recommendation is that it's each customer's decision to make."]

FIPS Mode

Apple macOS X 從10.8版開始的每個版本，都預設啟用FIPS Mode (送審到CMVP之安全政策文件，內文為Approved mode)，並且不能改變安全政策設定值，更不能關閉

```

Terminal Shell Edit View Window Help
aspens — zsh — 90x24
Last login: Sat Aug 20 08:26:53 on console
aspens@Mac-of-Aspen ~ % sudo /usr/libexec/cc_fips_test
Password:
Tracing: disabled
FIPSPST_USER [277249610428] fipspost_post:158: PASSED: (1 ms) - fipspost_post_integrity
FIPSPST_USER [277249662098] fipspost_post:164: PASSED: (0 ms) - fipspost_post_hmac
FIPSPST_USER [277249673184] fipspost_post:165: PASSED: (0 ms) - fipspost_post_aes_ecb
FIPSPST_USER [277249676609] fipspost_post:166: PASSED: (0 ms) - fipspost_post_aes_cbc
FIPSPST_USER [277252170906] fipspost_post:167: PASSED: (2 ms) - fipspost_post_rsa_sig
FIPSPST_USER [277252946318] fipspost_post:168: PASSED: (0 ms) - fipspost_post_ecdsa
FIPSPST_USER [277253200684] fipspost_post:169: PASSED: (0 ms) - fipspost_post_ecdh
FIPSPST_USER [277253237315] fipspost_post:170: PASSED: (0 ms) - fipspost_post_aes_ccm
FIPSPST_USER [277260709765] fipspost_post:172: PASSED: (7 ms) - fipspost_post_pbkdf
FIPSPST_USER [277260739714] fipspost_post:173: PASSED: (0 ms) - fipspost_post_kdf_ctr
FIPSPST_USER [277260747696] fipspost_post:174: PASSED: (0 ms) - fipspost_post_aes_gcm
FIPSPST_USER [277260752499] fipspost_post:175: PASSED: (0 ms) - fipspost_post_aes_xts
FIPSPST_USER [277260823427] fipspost_post:176: PASSED: (0 ms) - fipspost_post_tdes_cbc
FIPSPST_USER [277260837789] fipspost_post:177: PASSED: (0 ms) - fipspost_post_drbg_ctr
FIPSPST_USER [277260881854] fipspost_post:178: PASSED: (0 ms) - fipspost_post_drbg_hmac
FIPSPST_USER [277262927895] fipspost_post:180: PASSED: (2 ms) - fipspost_post_ffdh
FIPSPST_USER [277265420208] fipspost_post:181: PASSED: (2 ms) - fipspost_post_rsa_enc_dec
FIPSPST_USER [277265441887] fipspost_post:201: all tests PASSED (17 ms)
aspens@Mac-of-Aspen ~ %

```

https://csrc.nist.gov/CSRC/media/projects/cryp... 搜尋

9 頁, 共 30 頁 自動縮放

2.1.3 Tested Platforms

The module has been tested on the following platforms with and without AES-NI:

Manufacturer	Model	Operating System
Apple Inc.	Mac mini with i5 CPU	OS X 10.8
Apple Inc.	Mac with i7 CPU	OS X 10.8

Table 2: Tested Platforms

2.2 Modes of operation

The Apple OS X CoreCrypto Kernel Module, v3.0 has an Approved and non-Approved modes of operation. The Approved mode of operation is configured in the system by default and cannot be changed. If the device boots up successfully then CoreCrypto KEXT has passed all self-tests and is operating in the Approved mode. Any calls to the non-Approved security functions listed in Table 4 will cause the module to assume the non-Approved mode of operation.

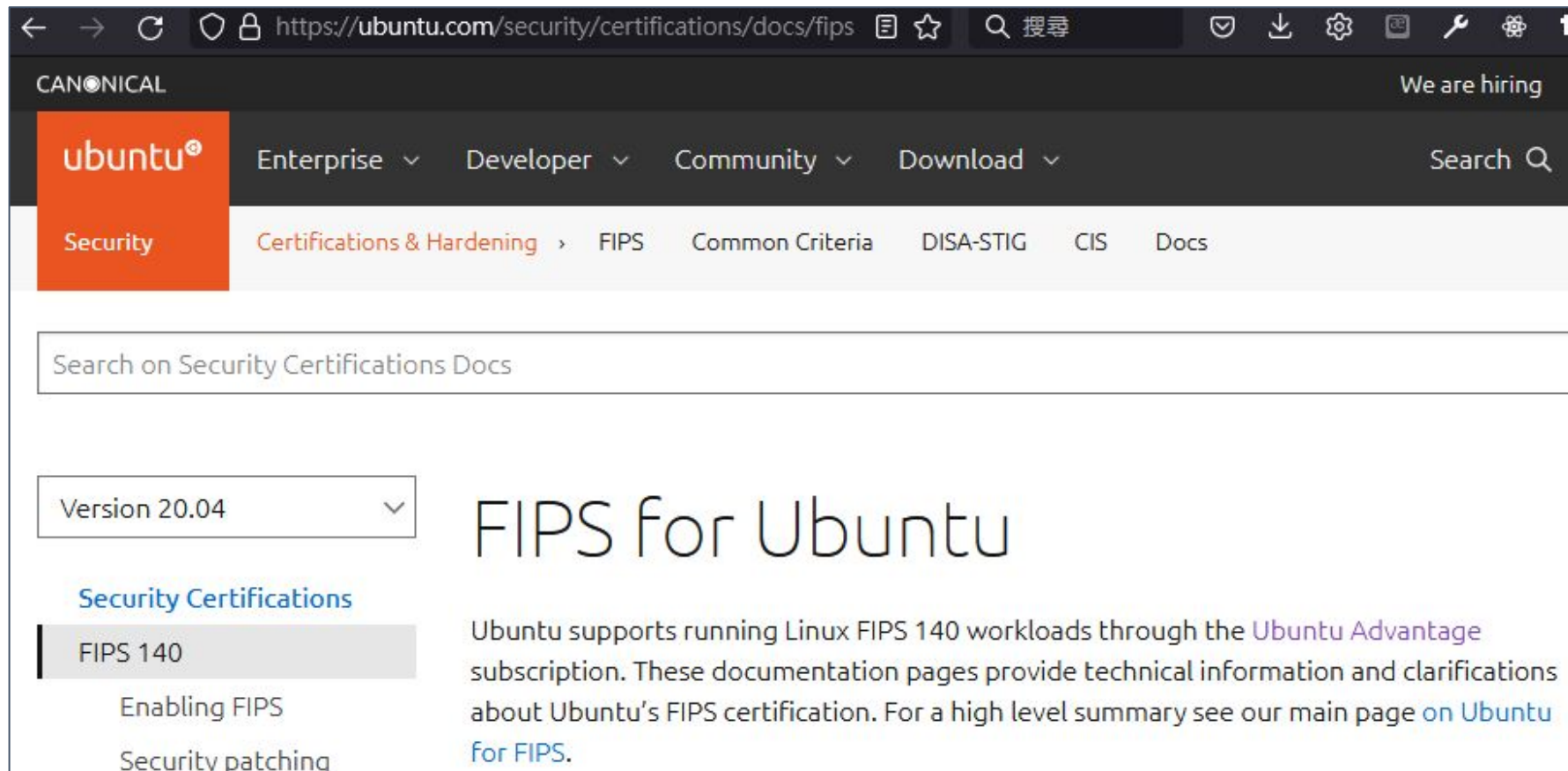
As all keys and Critical Security Parameters (CSP) handled by the module are ephemeral and there are no keys and CSPs shared between any functions, the module transitions back into FIPS mode immediately when invoking one of the approved ciphers. A re-invocation of the self-tests or integrity tests is not required.

Even when using this FIPS 140-2 non-approved mode, the module configuration ensures that the self-tests are always performed during initialization time of the module.

Last update: 2013-06-05 ©2013 Apple Inc.
 Version: 01.04 Document Id: FIPS_CORECRYPTO_OSX_KS_SECPOL_01.04 Page 9 of 30

FIPS Mode

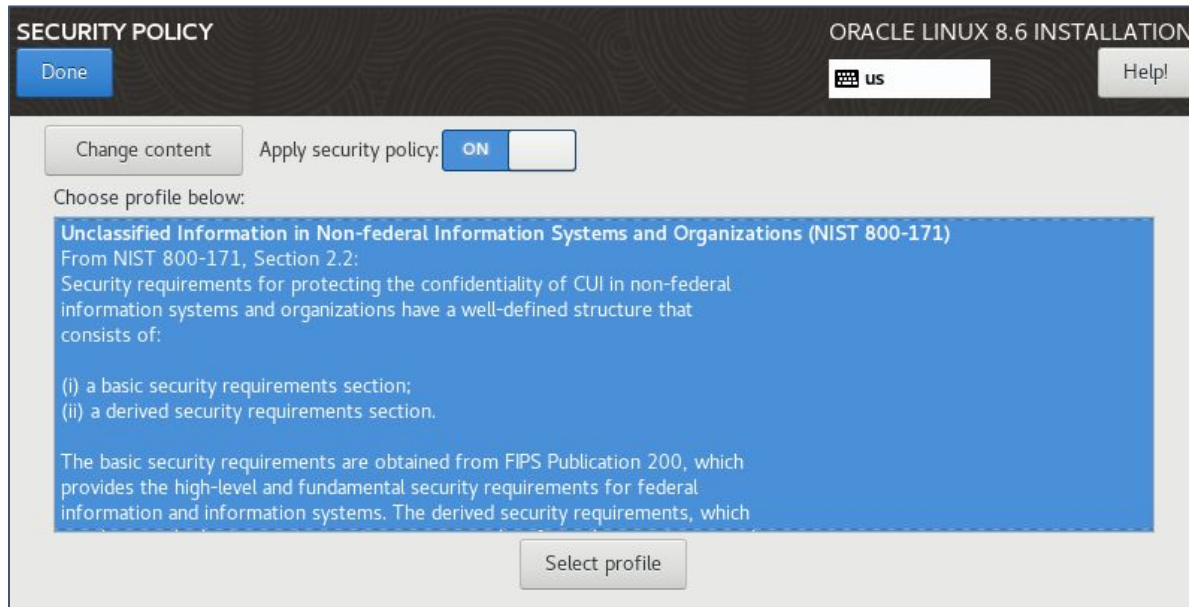
Ubuntu Linux需要訂閱Ubuntu Advantage才能啟用FIPS模式，只在LTS版本提供，也只有個人(personal)可以免費訂閱Ubuntu Advantage啟用FIPS模式



The screenshot shows a web browser window displaying the Ubuntu Security Certifications Docs page for FIPS. The browser address bar shows the URL <https://ubuntu.com/security/certifications/docs/fips>. The page header includes the Canonical logo, the Ubuntu logo, and navigation links for Enterprise, Developer, Community, and Download. A search bar is also present. The main navigation menu includes Security, Certifications & Hardening, FIPS, Common Criteria, DISA-STIG, CIS, and Docs. A search bar is located below the navigation menu. The page content includes a dropdown menu for Version 20.04, a section for Security Certifications with a sub-section for FIPS 140, and a main heading for FIPS for Ubuntu. The text below the heading states: "Ubuntu supports running Linux FIPS 140 workloads through the [Ubuntu Advantage](#) subscription. These documentation pages provide technical information and clarifications about Ubuntu's FIPS certification. For a high level summary see our main page [on Ubuntu for FIPS](#)."

FIPS Mode

美系Red Hat Enterprise Linux、Oracle Linux啟用FIPS模式，需要在「安裝之後，從OS內自行啟用」或是「安裝時套用特定安全政策」(圖例Oracle Linux 8.6套用NIST SP 800-171之安全政策：參照NIST FIPS 200、NIST SP 800-53實作設定)



```
[aspens@OracleLinux ~]$ fips-mode-setup --check  
FIPS mode is enabled.  
[aspens@OracleLinux ~]$ _
```


FIPS Mode

社群版的其他美系Linux目前在安裝時，若要選擇套用「特定安全政策」，則：

- Fedora無論是第幾版，一直都沒有「安全政策」能套用
- Rocky Linux 8.6 目前空有安全政策選項，按套用沒有執行
- CentOS 8.3 / 8-Stream、CentOS 7.9的2020年9月版(2009)，沒有安全政策能套用
- CentOS 7.9的2022年7月-2版(2207-02)，只有基本安全政策能套用，然而基本安全政策沒有開啟FIPS模式

FIPS Mode

行動作業系統：

- Android和Apple iOS預設都啟用FIPS模式
- Android只有Google、Samsung、Motorola、Avaya等幾家大廠，有將BoringCrypto密碼學模組送驗通過CMVP審查為FIPS 140-2 Level 1
- 目前開發與政府單位有關的「行動應用App」會被要求遵守**行動資安聯盟** (<https://www.mas.org.tw/download/app>) 訂定之「**行動應用App基本資安規範**」並且將App送驗至第三方實驗室
- 前身為經濟部工業局訂定之規範，即要求使用FIPS合規之密碼學函式



https://www.mas.org.tw/storage/files/2/original/60794333623d7234c692f.pdf 搜尋

13 頁，共 32 頁 自動縮放

3.37. 安全亂數產生函式 (Secure Random Number Generator)

符合或引用 ANSI X9.17、FIPS 140-2、NIST SP 800-22 以及 SP 800-90A (CAVP Testing: Random Number Generators) 至少其中一項標準之亂數產生函式。

3.38. 安全網域 (Secure Domain)

範圍包括開發商、客戶所屬網域或一般熟知之公共安全網域，一般熟知之公共安全網域包括 Facebook、Google 或 Twitter 等支援 OAuth 2.0 協定之應用。

3.39. 安全加密函式 (Secure Encryption Function)

符合 FIPS 140-2 Annex A 之加密函式。

3.40. 系統憑證儲存設施(System Credentials Storage Facilities)

指行動作業系統提供行動應用程式開發人員及行動裝置使用者用於儲存用戶憑證或密碼金鑰之服務，例如：Keystore (Android)、Keychain (iOS) 或其它類似機制。

Cryptographic Failures

密碼學問題已經是
Top#2的網站應用程式缺陷

https://csrc.nist.gov/Projects/Cryptographic-Algorithm-Validation-Program

適應性 640 x 1080 DPR: 1 不限速 UA: 自訂 User Agent

Currently, CAVP tests the following cryptographic algorithms*. Follow the links to **algorithm specifications**, **validation testing requirements**, **validation lists** and **test vectors**.

Block Ciphers	AES , Triple DES , Skipjack (decryption only) Tests for ECB, CBC, CFB and OFB modes.
Block Cipher Modes	CCM , CMAC , GCM / GMAC / XPN , Key Wrap , XTS
Digital Signatures	FIPS 186-4: DSA , ECDSA , RSA FIPS 186-2: DSA , ECDSA , RSA
Key Derivation Functions	KBKDF
Key Management	KAS
Message Authentication	HMAC (FIPS 198-1)
Random Number Generation	DRBG
Secure Hashing	SHA-2 , SHA-1 SHA-3
Component Testing	ECC-CDH (SP 800-56A), ECDSA Signature (FIPS 186-4), KDF (SP 800-135), RSA PKCS1-v1.5 RSASP1 (FIPS 186-4), RSA PKCS1-vPSS RSASP1 (FIPS 186-4), RSADP Decryption (SP 800-56B; PKCS#1 v2.1)

https://owasp.org/Top10/

Home

OWASP Top 10:2021

Home

Notice

Introduction

How to use the OWASP Top 10 as a standard

How to start an AppSec program with the OWASP Top 10

About OWASP

Top 10:2021 List

A01 Broken Access Control

A02 Cryptographic Failures

A03 Injection

A04 Insecure Design

A05 Security Misconfiguration

A06 Vulnerable and Outdated Components

A07 Identification and Authentication Failures

A08 Software and Data Integrity Failures

A09 Security Logging and Monitoring Failures

A10 Server Side Request Forgery (SSRF)

https://cwe.mitre.org/data/definitions/1346.html



Home > CWE List > CWE- Individual Dictionary Definition (4.8)

ID Lookup: Go

Home | About | CWE List | Scoring | Mapping Guidance | Community | News | Search

CWE CATEGORY: OWASP Top Ten 2021 Category A02:2021 - Cryptographic Failures

Category ID: 1346

Summary

Weaknesses in this category are related to the A02 category "Cryptographic Failures" in the OWASP Top Ten 2021.

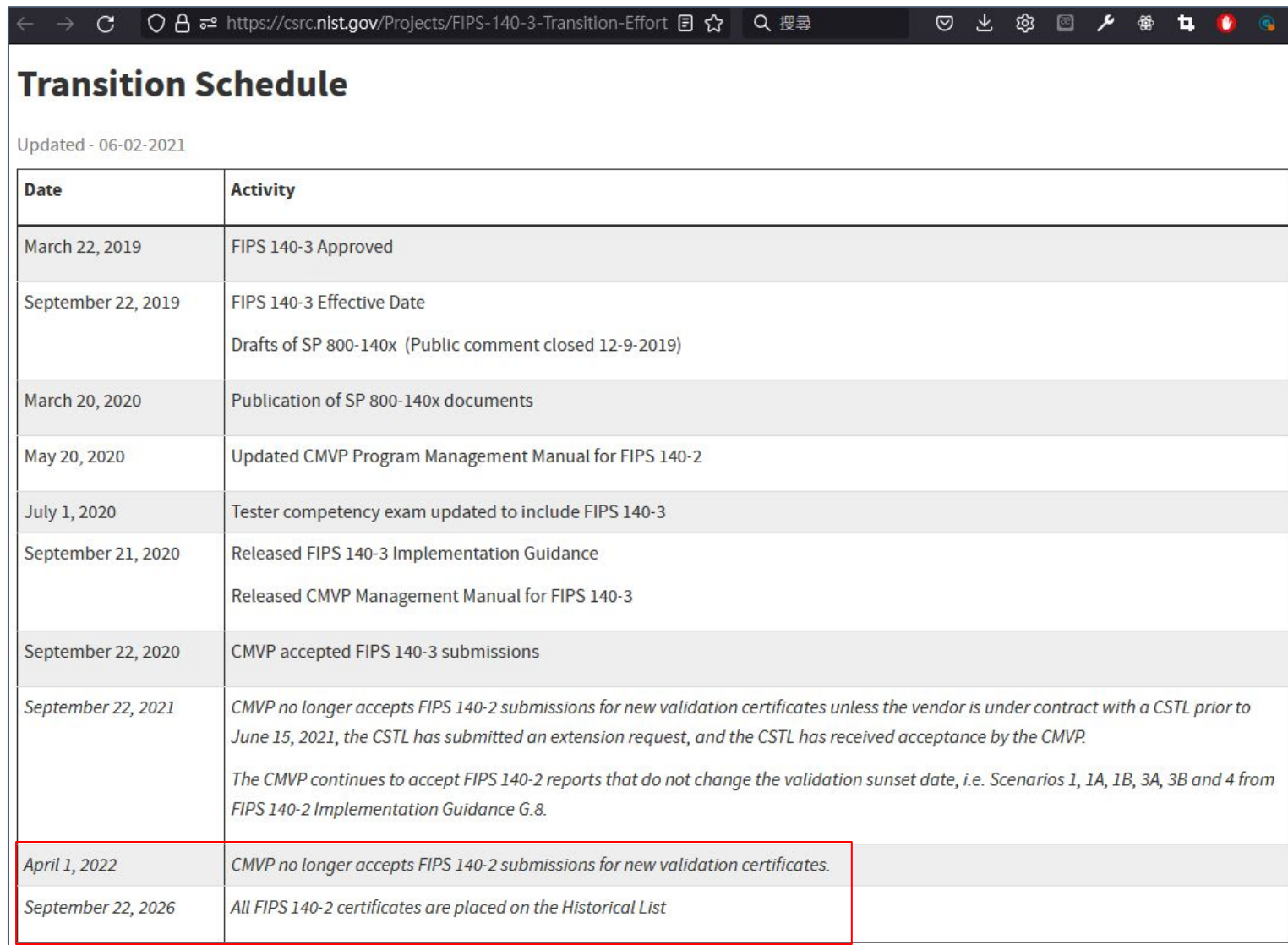
Membership

Nature	Type	ID	Name
MemberOf	V	1344	Weaknesses in OWASP Top Ten (2021)
HasMember	B	261	Weak Encoding for Password
HasMember	B	296	Improper Following of a Certificate's Chain of Trust
HasMember	C	310	Cryptographic Issues
HasMember	B	319	Cleartext Transmission of Sensitive Information
HasMember	V	321	Use of Hard-coded Cryptographic Key
HasMember	B	322	Key Exchange without Entity Authentication
HasMember	V	323	Reusing a Nonce, Key Pair in Encryption
HasMember	B	324	Use of a Key Past its Expiration Date
HasMember	B	325	Missing Cryptographic Step
HasMember	C	326	Inadequate Encryption Strength
HasMember	C	327	Use of a Broken or Risky Cryptographic Algorithm
HasMember	B	328	Use of Weak Hash
HasMember	V	329	Generation of Predictable IV with CBC Mode
HasMember	C	330	Use of Insufficiently Random Values
HasMember	B	331	Insufficient Entropy
HasMember	B	335	Incorrect Usage of Seeds in Pseudo-Random Number Generator (PRNG)
HasMember	V	336	Same Seed in Pseudo-Random Number Generator (PRNG)
HasMember	V	337	Predictable Seed in Pseudo-Random Number Generator (PRNG)
HasMember	B	338	Use of Cryptographically Weak Pseudo-Random Number Generator (PRNG)
HasMember	C	340	Generation of Predictable Numbers or Identifiers
HasMember	B	347	Improper Verification of Cryptographic Signature
HasMember	B	523	Unprotected Transport of Credentials
HasMember	C	720	OWASP Top Ten 2007 Category A9 - Insecure Communications
HasMember	B	757	Selection of Less-Secure Algorithm During Negotiation ('Algorithm Downgrade')
HasMember	V	759	Use of a One-Way Hash without a Salt
HasMember	V	760	Use of a One-Way Hash with a Predictable Salt
HasMember	V	780	Use of RSA Algorithm without OAEP
HasMember	C	818	OWASP Top Ten 2010 Category A9 - Insufficient Transport Layer Protection
HasMember	B	916	Use of Password Hash With Insufficient Computational Effort

FIPS 140-2 Sunset

目前CMVP已經不再接受
密碼學模組的提交
新的FIPS 140-2驗證

已通過FIPS 140-2驗證的
密碼學模組將在**四年**後全數落日



Transition Schedule

Updated - 06-02-2021

Date	Activity
March 22, 2019	FIPS 140-3 Approved
September 22, 2019	FIPS 140-3 Effective Date Drafts of SP 800-140x (Public comment closed 12-9-2019)
March 20, 2020	Publication of SP 800-140x documents
May 20, 2020	Updated CMVP Program Management Manual for FIPS 140-2
July 1, 2020	Tester competency exam updated to include FIPS 140-3
September 21, 2020	Released FIPS 140-3 Implementation Guidance Released CMVP Management Manual for FIPS 140-3
September 22, 2020	CMVP accepted FIPS 140-3 submissions
September 22, 2021	<i>CMVP no longer accepts FIPS 140-2 submissions for new validation certificates unless the vendor is under contract with a CSTL prior to June 15, 2021, the CSTL has submitted an extension request, and the CSTL has received acceptance by the CMVP.</i> <i>The CMVP continues to accept FIPS 140-2 reports that do not change the validation sunset date, i.e. Scenarios 1, 1A, 1B, 3A, 3B and 4 from FIPS 140-2 Implementation Guidance G.8.</i>
April 1, 2022	<i>CMVP no longer accepts FIPS 140-2 submissions for new validation certificates.</i>
September 22, 2026	<i>All FIPS 140-2 certificates are placed on the Historical List</i>

Transition of Cryptographic Module



目前主流OS都宣布預備支援或已支援FIPS 140-3, 待實驗室測試後, 將由NIST CMVP核可為**FIPS 140-3 Validated**

https://support.apple.com/zh-tw/guide/sccc/sccc5eb3dc4fa/web

FIPS 140-3 認證

2020 年, Apple 發表了以 Apple 晶片為基礎的 Mac 電腦。下表的「模組資訊」欄指出了加密編譯模組對 Apple 晶片或採用 Intel 架構的 Mac 電腦的適用性。

【注意】許多採用 Intel 架構的 Mac 電腦都包含 Apple T2 安全晶片。如需 T2 晶片認證的相關資訊, 請參閱 [Apple T2 安全晶片的的安全性認證](#)。

macOS ssh 用戶端

OpenSSH 可設定為使用 FIPS 140-3 已驗證模組來處理特定 FIPS 140-3 演算法。組織可以執行 Apple 所提供經簽署與公證的安裝程式 (密碼為 FIPS140Mode)。安裝程式會在 Mac 上放置兩個檔案:

- fips_ssh_config: 位於 /private/etc/ssh/ssh_config.d/
- fips_sshd_config: 位於 /private/etc/ssh/sshd_config.d/

macOS 接著會使用這些檔案, 將 OpenSSH 適用的加密方式限制為僅受 NIST 驗證的加密方式, 並確保 OpenSSH 用戶端是使用由平台提供且經過驗證的加密編譯模組。管理者也可以製作自己的檔案。如需更多資訊, 請參閱 macOS 12.0.1 或以上版本的 `apple_ssh_and_fips` man 頁面。

目前狀態

macOS 11 Big Sur 使用者空間、核心空間和 Secure Key Store 已完成實驗室測試, 並已由實驗室推薦給 CMVP 進行驗證。它們列於 [檢測中的模組列表 \(Modules in Process List\)](#)。

macOS 12 Monterey 使用者空間、核心空間和 Secure Key Store 正在接受實驗室測試。它們列於 [實作待測列表 \(Implementation Under Test List\)](#) 中。

日期	憑證/文件	模組資訊
作業系統發佈日期: 2021	憑證: 尚未通過認證	標題: Apple Corecrypto 模組 v12.0
驗證日期: —	文件: 憑證 安全性規則 Crypto Officer 指引	作業系統: Apple 晶片上的 macOS 12 Monterey 環境: Apple 晶片、使用者、軟體 類型: 軟體 安全性層級: 1

https://access.redhat.com/articles/2918071

FIPS 140-2 and FIPS 140-3

Federal Information Processing Standard 140-2 and 140-3 ensures that cryptographic tools implement their algorithms properly. There are a number of FIPS 140-2-related articles in the [Red Hat Customer Portal](#). You'll find a complete list of all FIPS 140-2 and FIPS 140-3 certificates at the [NIST CMVP website](#). The Red Hat certificates are below.

A note on applicability: The exact platform and environment tested is specified in the Security Policy for each certificate, though generally applicable to other Red Hat products where the binary versions of modules are running unmodified as well. FIPS 140 certificates issued to Red Hat are not generally applicable to non-Red Hat products. Please see the Security Policy, available at the links that follow, for specifics. Module binaries may be unchanged across Red Hat Enterprise Linux minor releases. In this case Red Hat reports the same applicable module version and certificate for such releases.

Red Hat Enterprise Linux 9.0

Cryptographic Module	Module Version	Associated Packages	Validation Status	Certificate
OpenSSL	TBD	TBD	Implementation Under Test	N/A
Libgcrypt	TBD	TBD	Implementation Under Test	N/A
Kernel Cryptographic API	TBD	TBD	Implementation Under Test	N/A

https://ubuntu.com/security/fips

FIPS 140-3 and Ubuntu

In September 2021, NIST began phasing out FIPS 140-2. Certifications under FIPS 140-2 remain valid no longer than September 2026 and new products are expected to be certified under FIPS 140-3. FIPS 140-3 is a combined effort of NIST and ISO with the Security and Testing requirements for cryptographic modules being published as ISO/IEC 19790 and ISO/IEC 24759. Canonical is preparing Ubuntu for the new certification, and intends to provide FIPS 140-3 certified cryptographic packages on a future LTS release of Ubuntu.

https://rockylinux.org/news/certifications-fips-2022-06-11/

JUNE 11, 2022

FIPS Validation Update - June 2022

THANK YOU
CIQ
<http://www.ciq.co>

We are excited to announce that Rocky Linux has reached a significant step in the FIPS 140-3 validation process; right on schedule, Rocky Linux is now named in the [NIST Implementation Under Test List](#).

A big, gigantic thank you to our founding partner and sponsor [@CtrlIQ](#) (CIQ), who has arranged and paid for the FIPS validation process and will be providing it back the entire RESF / Rocky community for free!

4

Introduction of NIST FIPS 140-3

FIPS 140-3

「 Announcing the Standard for **SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES** 」

目的

分級

11大項要求

3. Explanation. This standard specifies the security requirements that will be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information (hereafter referred to as sensitive information). The standard provides four increasing, qualitative levels of security: Level 1, Level 2, Level 3, and Level 4. These levels are intended to cover the wide range of potential applications and environments in which cryptographic modules may be employed. The security requirements cover areas related to the secure design and implementation of a cryptographic module. These areas include cryptographic module specification; cryptographic module interfaces; roles, services, and authentication; software/firmware security; operating environment; physical security; non-invasive security; sensitive security parameter management; self-tests; life-cycle assurance; and mitigation of other attacks. This standard supersedes FIPS 140-2, *Security Requirements for Cryptographic Modules*, in its entirety.

FIPS 140-3

FIPS 140-3 Process

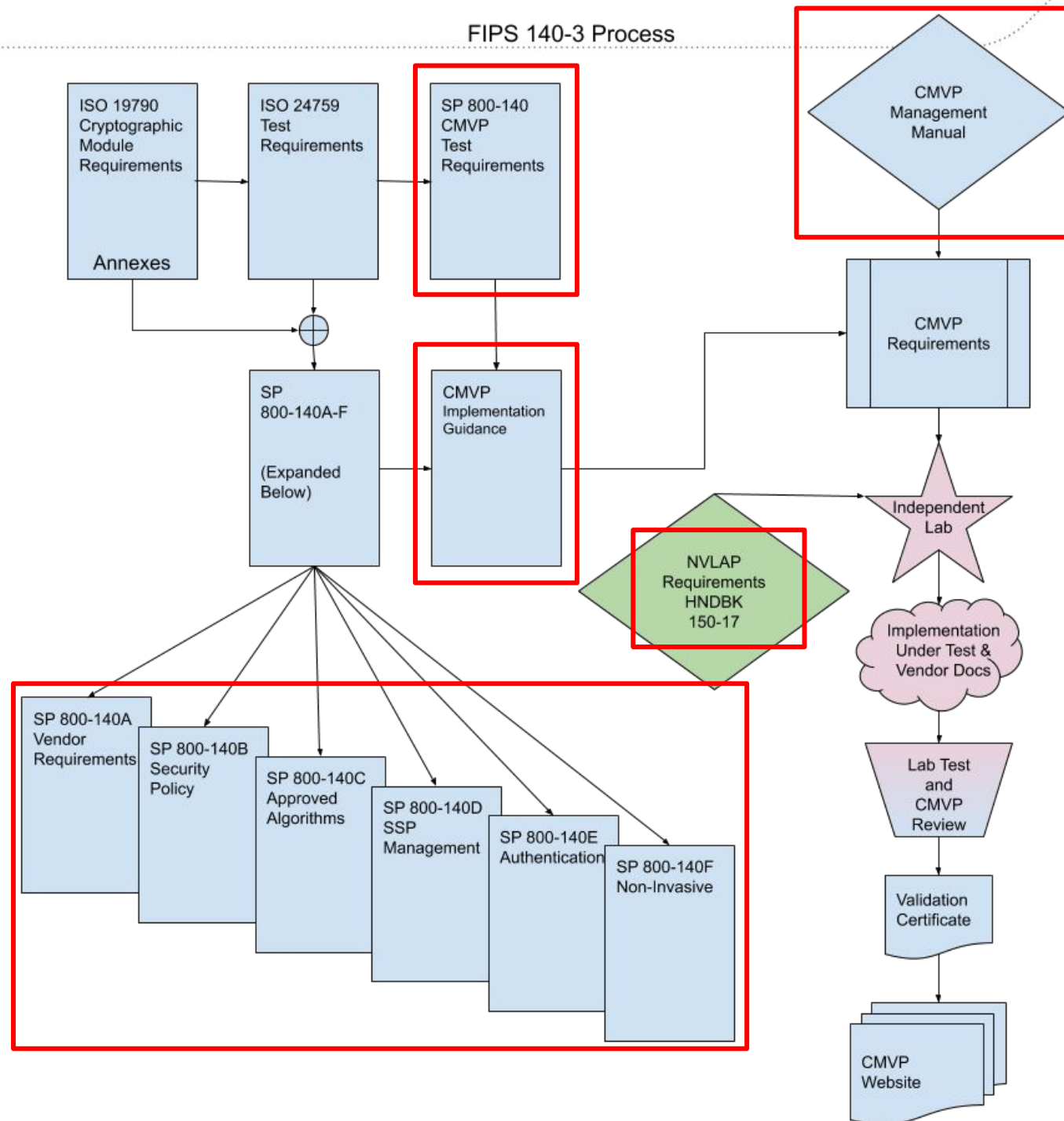


ISO/IEC 19790:2012
Information technology – Security techniques – Security requirements for cryptographic modules

ISO/IEC 24759:2017
Information technology – Security techniques – Test requirements for cryptographic modules

Table 1: NIST SPs that Modify ISO/IEC Standards

NIST Special Publication		ISO/IEC 19790:2012(E)	ISO/IEC 24759:2017(E)
SP 800-140		--	§6.1 through §6.12
SP 800-140A	modifies	Annex A	§6.13
SP 800-140B		Annex B	§6.14
SP 800-140C		Annex C	§6.15
SP 800-140D		Annex D	§6.16
SP 800-140E		Annex E	§6.17
SP 800-140F		Annex F	§6.18



FIPS 140-3 密碼學模組驗證程序管理手冊(Draft)

參與角色

供應者:

設計與提供密碼學模組

實驗室:

檢測與評估密碼學模組

驗證小組:

認可密碼學模組之驗證

使用者:

指定與採購核可之密碼學
模組, 使資安獲得確保

The CMVP will review the CST laboratory's position and rationale supporting its conclusion. If the CMVP concurs that the official request is without merit, no further action is taken. If the CMVP concurs that the official request has merit, a security risk assessment will be performed regarding the non-conformance issue. Please see [Validated Module Issue Assessment Process](#) for the flow diagram to the assessment process.

2.5 Roles and Responsibilities of Program Participants

The various roles and responsibilities of the participants in the CMVP are illustrated in Figure 1 below.

Who	Vendor	CST Laboratory	CMVP	User
Function	Designs & Produces	Tests for Conformance	Reviews & Approves	Specifies & Purchases
Output	Cryptographic Modules	Assessment Report	Validation List	Security with Assurance

Figure 1- Roles, Responsibilities, and Output in the CMVP Process

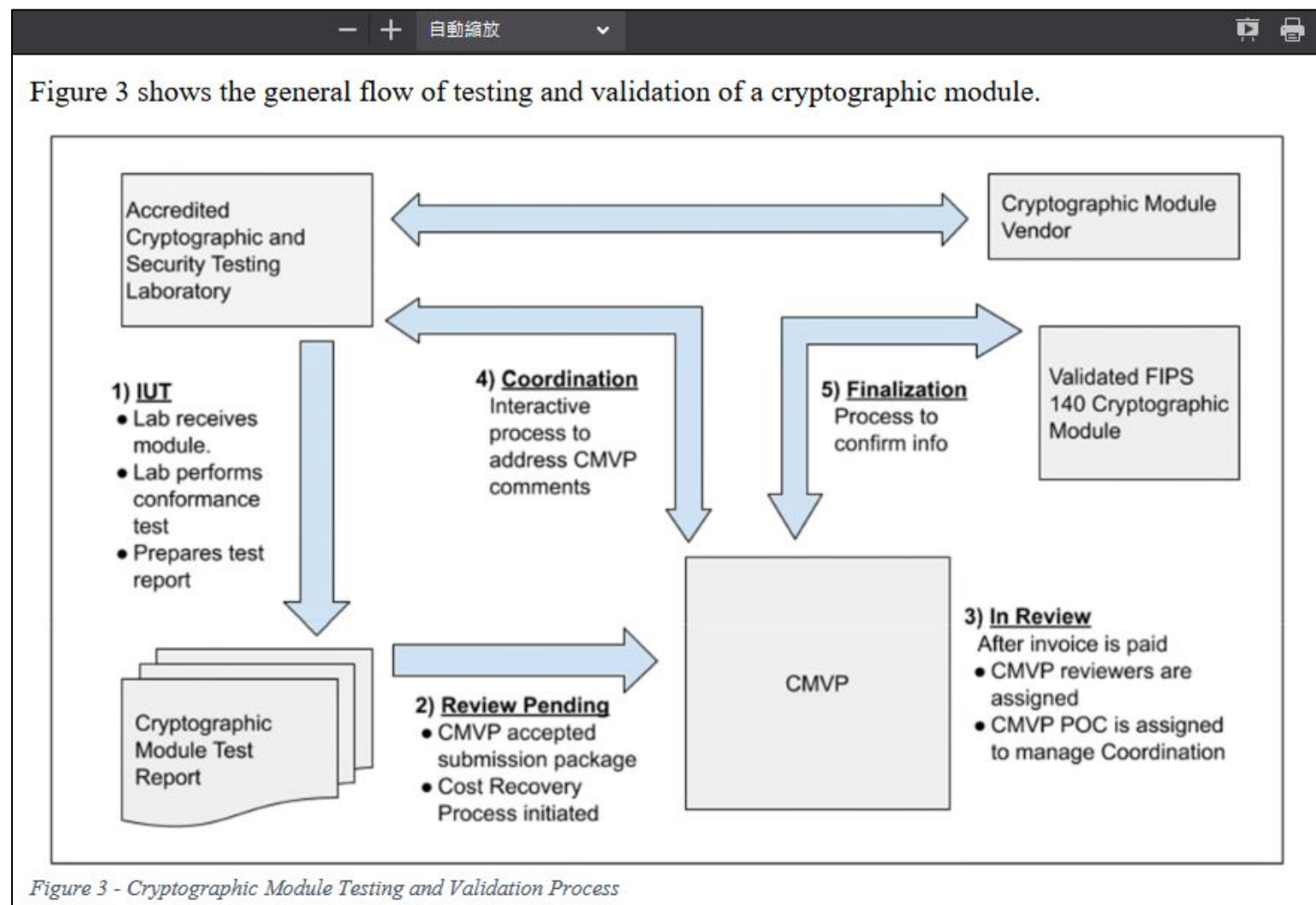
FIPS 140-3 密碼學模組驗證程序管理手冊(Draft)

送驗流程

已驗證的密碼學模組一旦驗出弱點

CMVP 認證就失效

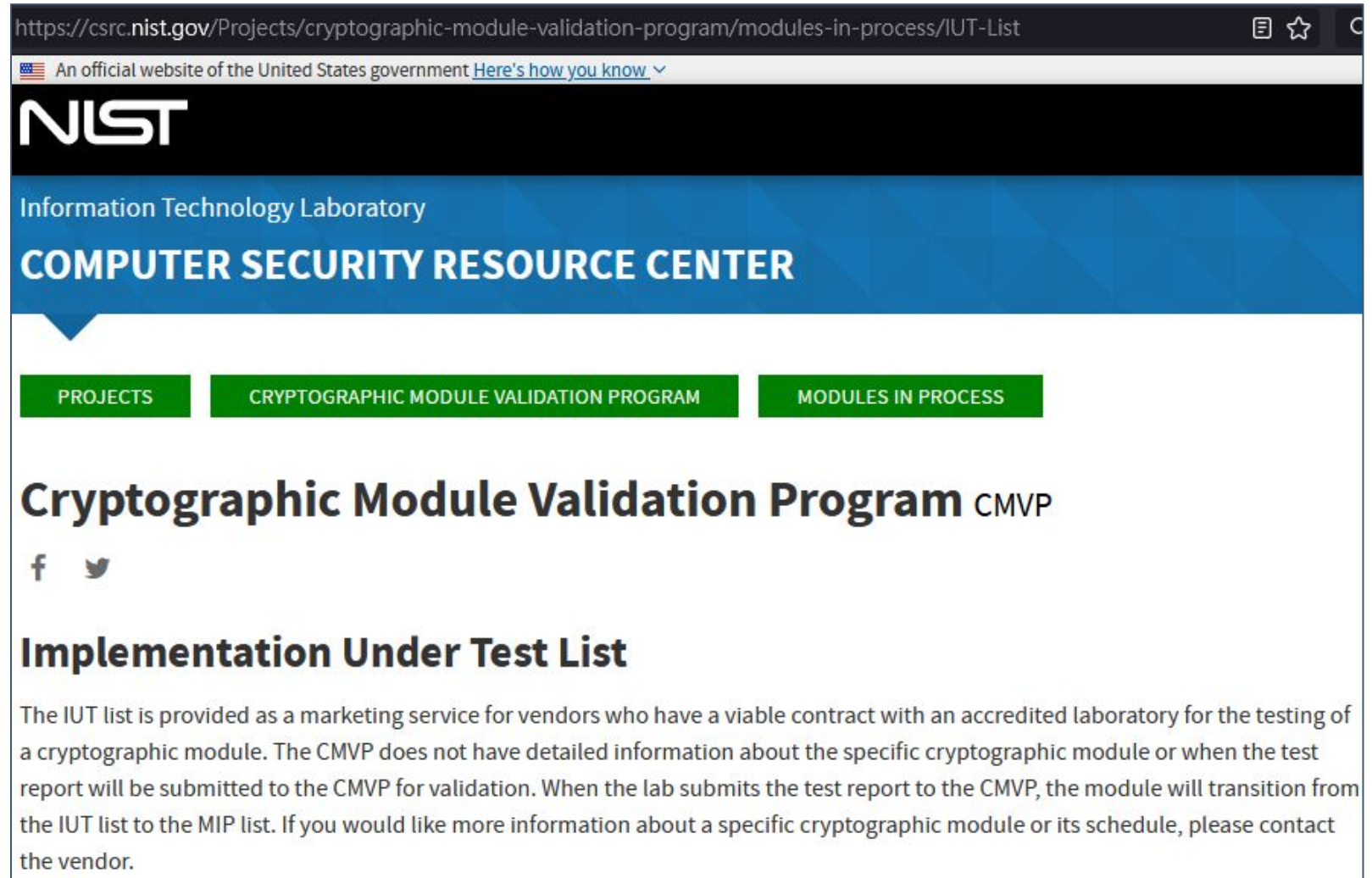
需要修補弱點後重新進行驗證流程



FIPS 140-3 密碼學模組驗證程序 IUT List

(1) IUT

已經送驗 FIPS 140-3正在實驗室進行驗證並且由實驗室產出驗證評定報告



The screenshot shows the NIST website page for the Cryptographic Module Validation Program (CMVP) IUT List. The URL is <https://csrc.nist.gov/Projects/cryptographic-module-validation-program/modules-in-process/IUT-List>. The page features the NIST logo and the text "Information Technology Laboratory" and "COMPUTER SECURITY RESOURCE CENTER". There are three green buttons: "PROJECTS", "CRYPTOGRAPHIC MODULE VALIDATION PROGRAM", and "MODULES IN PROCESS". The main heading is "Cryptographic Module Validation Program CMVP" with social media icons for Facebook and Twitter. Below this is the section "Implementation Under Test List" with a paragraph of text explaining the IUT list as a marketing service for vendors.

<https://csrc.nist.gov/Projects/cryptographic-module-validation-program/modules-in-process/IUT-List>

An official website of the United States government [Here's how you know](#)

NIST
Information Technology Laboratory
COMPUTER SECURITY RESOURCE CENTER

PROJECTS CRYPTOGRAPHIC MODULE VALIDATION PROGRAM MODULES IN PROCESS

Cryptographic Module Validation Program CMVP

f t

Implementation Under Test List

The IUT list is provided as a marketing service for vendors who have a viable contract with an accredited laboratory for the testing of a cryptographic module. The CMVP does not have detailed information about the specific cryptographic module or when the test report will be submitted to the CMVP for validation. When the lab submits the test report to the CMVP, the module will transition from the IUT list to the MIP list. If you would like more information about a specific cryptographic module or its schedule, please contact the vendor.

2022/Sep/19

列表上已有56家廠牌
共137項產品送驗

FIPS 140-3 密碼學模組驗證程序 MIP List

(2) Review Pending

(3) In Review

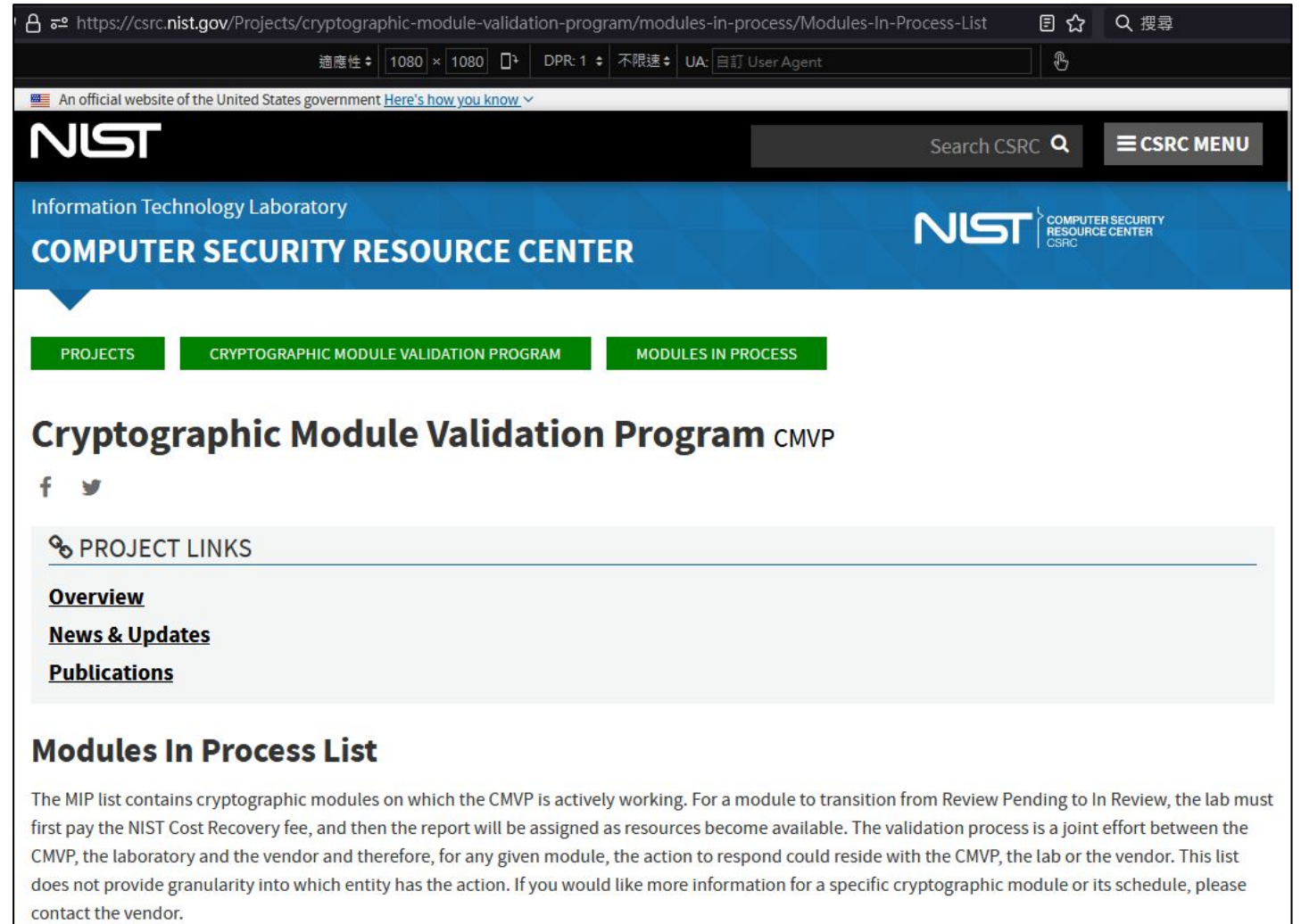
(4) Coordination

(5) Finalization

實驗室已經完成驗證，
正在CMVP處理中

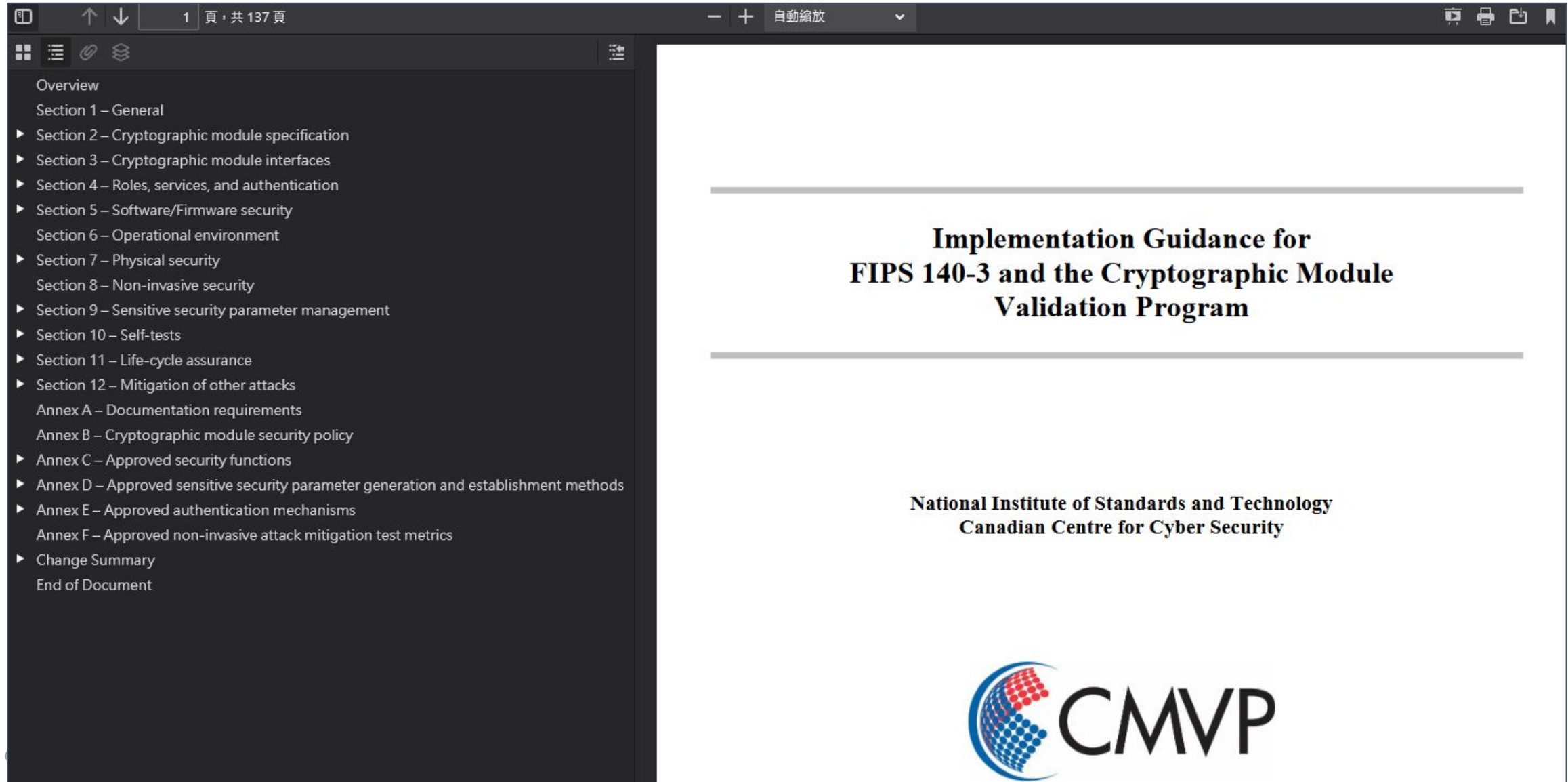
2022/Sep/19

列表包含FIPS 140-2提交截止日
之前提交的密碼學模組，目前列表上
有67項已完成FIPS 140-3送驗，正
在進行審定作業的密碼學模組



The screenshot shows the NIST CSRC website page for the Cryptographic Module Validation Program (CMVP) Modules In Process List. The page includes the NIST logo, the CSRC logo, and navigation links for PROJECTS, CRYPTOGRAPHIC MODULE VALIDATION PROGRAM, and MODULES IN PROCESS. The main heading is "Cryptographic Module Validation Program CMVP". Below this, there are social media icons for Facebook and Twitter, and a section for "PROJECT LINKS" with links for Overview, News & Updates, and Publications. The "Modules In Process List" section is partially visible, with a paragraph explaining that the list contains cryptographic modules on which the CMVP is actively working, and that the validation process is a joint effort between the CMVP, the laboratory, and the vendor.

FIPS 140-3 實作指南, 修改自ISO/IEC19790:2012



1 頁, 共 137 頁


自動縮放

Overview

- Section 1 – General
- ▶ Section 2 – Cryptographic module specification
- ▶ Section 3 – Cryptographic module interfaces
- ▶ Section 4 – Roles, services, and authentication
- ▶ Section 5 – Software/Firmware security
- Section 6 – Operational environment
- ▶ Section 7 – Physical security
- Section 8 – Non-invasive security
- ▶ Section 9 – Sensitive security parameter management
- ▶ Section 10 – Self-tests
- ▶ Section 11 – Life-cycle assurance
- ▶ Section 12 – Mitigation of other attacks
- Annex A – Documentation requirements
- Annex B – Cryptographic module security policy
- ▶ Annex C – Approved security functions
- ▶ Annex D – Approved sensitive security parameter generation and establishment methods
- ▶ Annex E – Approved authentication mechanisms
- Annex F – Approved non-invasive attack mitigation test metrics
- ▶ Change Summary
- End of Document

Implementation Guidance for FIPS 140-3 and the Cryptographic Module Validation Program

**National Institute of Standards and Technology
Canadian Centre for Cyber Security**



FIPS 140-2 與 FIPS 140-3 資安要求差異

FIPS 140-2

1. **密碼學模組規格**
(Cryptographic Module Specification)
2. **密碼學模組埠口與介面**
(Cryptographic Module Ports and Interfaces)
3. **角色、服務與身分認證**
(Roles, Services, and Authentication)
4. **有限狀態模型**
(Finite State Model)
5. **物理性資安防護**
(Physical Security)
6. **操作環境**
(Operational Environment)

FIPS 140-3

1. **密碼學模組規格**
(Cryptographic Module Specification)
2. **密碼學模組介面**
(Cryptographic Module Interfaces)
3. **角色、服務與身分認證**
(Roles, Services, and Authentication)
4. **軟韌體資安**
(Software/Firmware Security)
5. **作業環境**
(Operating Environment)
6. **物理性資安防護**
(Physical Security)

FIPS 140-2 與 FIPS 140-3 資安要求差異 (cont.)

FIPS 140-2

- 7. 密碼學金鑰管理**
(Cryptographic Key Management)
- 8. 電磁干擾 / 電磁相容性**
(Electromagnetic Interference /
Electromagnetic Compatibility)
- 9. 自我測試**
(Self-Tests)
- 10. 設計面確保**
(Design Assurance)
- 11. 緩解其他攻擊方式**
(Mitigation of Other Attacks)

FIPS 140-3

- 7. 非侵入式資安**
(Non-invasive Security)
- 8. 敏感安全參數管理**
(Sensitive Security Parameter Management)
- 9. 自我測試**
(Self-Tests)
- 10. 生命週期確保**
(Life-cycle Assurance)
- 11. 緩解其他攻擊方式**
(Mitigation of Other Attacks)

FIPS 140-2 與 FIPS 140-3 附件標題差異

FIPS 140-2 附錄(Appendix)與附件(Annex)標題

Appendix A: Summary Of **Documentation Requirements**

Appendix B: Recommended Software Development Practices

Appendix C: Cryptographic Module **Security Policy**

Appendix D: Selected Bibliography

Appendix E: Applicable Internet Uniform Resource Locators (URL)

Annex A: **Approved Security Functions**

Annex B: Approved Protection Profiles

Annex C: Approved Random Number Generators

Annex D: Approved Key Establishment Techniques

FIPS 140-3 附件標題

800-140: Derived Test Requirements

800-140A: **Documentation Requirements**

800-140B: **Security Policy** Requirements

800-140C: **Approved Security Functions**

800-140D: Approved **Sensitive Parameter** Generation and Establishment Methods

800-140E: Approved Authentication Mechanisms

800-140F: Approved Non-Invasive Attack Mitigation Test Metrics

FIPS 140-2 與 FIPS 140-3 重要術語差異

FIPS 140-2

- 亂數產生器
(Random Number Generators, RNG)
- 決定性亂數產生器
(Deterministic Random Number Generators, DRNG)
- 非決定性亂數產生器
(Non-Deterministic Random Number Generator, NDRNG)

FIPS 140-3

- 隨機位元產生器
(Random Bit Generators, RBG)
- 決定性隨機位元產生器
(Deterministic Random Bit Generators, DRBG)
- 熵 (Entropy)
- 不再進行「連續亂數產生測試」
(Continuous Random Number Generator Tests, CRNGT)

FIPS 140-3 密碼學模組驗證程序

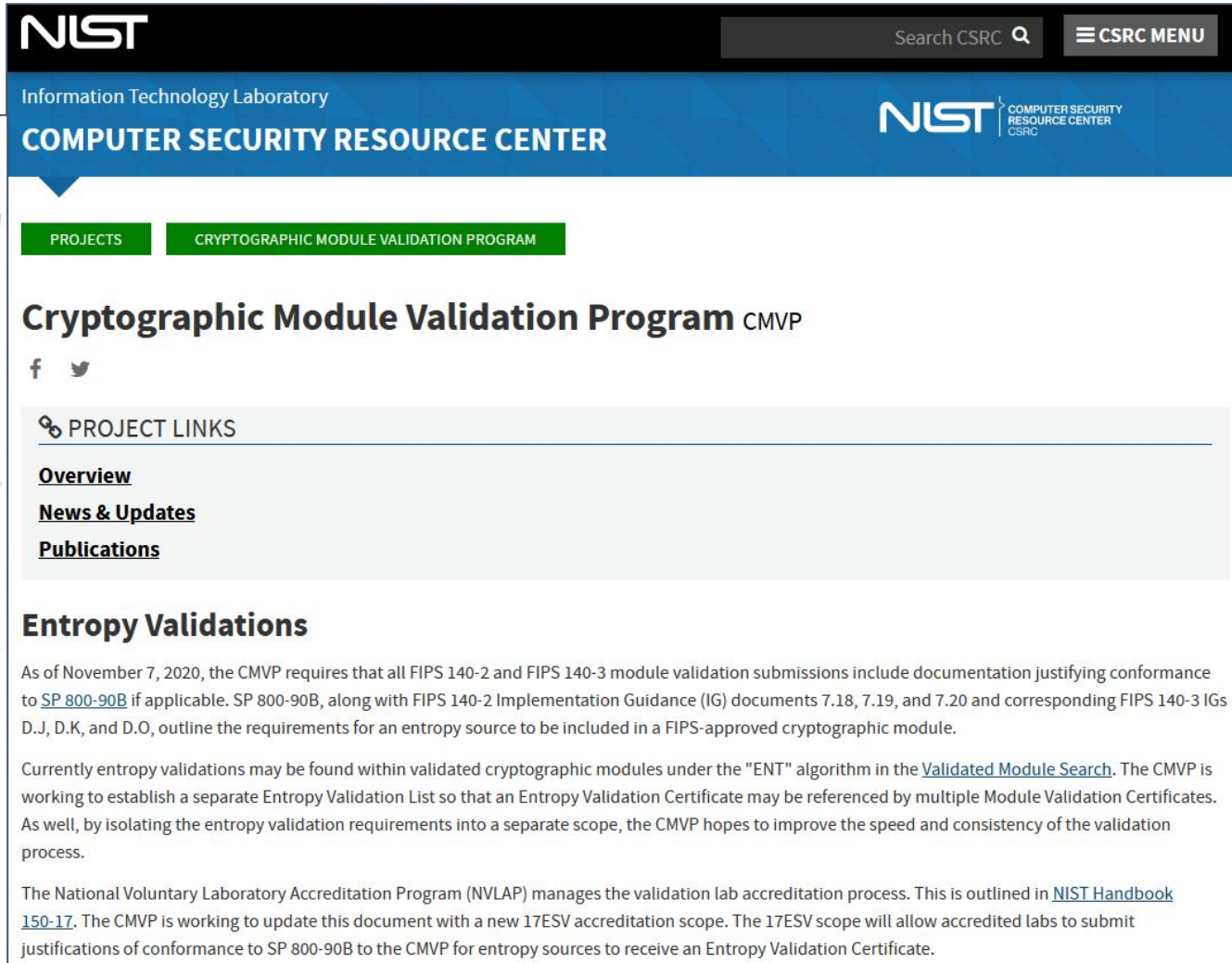
不只密碼學演算法，連**Entropy**也要驗證

NIST Special Publication 800-90B

Recommendation for the Entropy Sources Used for Random Bit Generation

Meltem Sönmez Turan
Elaine Barker
John Kelsey
Kerry A. McKay
Mary L. Baish
Mike Boyle

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.SP.800-90B>



NIST Information Technology Laboratory
COMPUTER SECURITY RESOURCE CENTER

Search CSRC 🔍 CSRC MENU

PROJECTS CRYPTOGRAPHIC MODULE VALIDATION PROGRAM

Cryptographic Module Validation Program CMVP

f t

PROJECT LINKS

- [Overview](#)
- [News & Updates](#)
- [Publications](#)

Entropy Validations

As of November 7, 2020, the CMVP requires that all FIPS 140-2 and FIPS 140-3 module validation submissions include documentation justifying conformance to [SP 800-90B](#) if applicable. SP 800-90B, along with FIPS 140-2 Implementation Guidance (IG) documents 7.18, 7.19, and 7.20 and corresponding FIPS 140-3 IGS D.J, D.K, and D.O, outline the requirements for an entropy source to be included in a FIPS-approved cryptographic module.

Currently entropy validations may be found within validated cryptographic modules under the "ENT" algorithm in the [Validated Module Search](#). The CMVP is working to establish a separate Entropy Validation List so that an Entropy Validation Certificate may be referenced by multiple Module Validation Certificates. As well, by isolating the entropy validation requirements into a separate scope, the CMVP hopes to improve the speed and consistency of the validation process.

The National Voluntary Laboratory Accreditation Program (NVLAP) manages the validation lab accreditation process. This is outlined in [NIST Handbook 150-17](#). The CMVP is working to update this document with a new 17ESV accreditation scope. The 17ESV scope will allow accredited labs to submit justifications of conformance to SP 800-90B to the CMVP for entropy sources to receive an Entropy Validation Certificate.

FIPS 140-3 新增之術語

1. 密碼學模組規格 (Cryptographic Module Specification)

- 一般操作模式 (Normal Operation) 自我測試成功, 提供可以設定或不可設定, 完整的「演算法、安全函式、服務或程序」
- 降階操作模式 (Degraded Operation) 操作前執行的自我測試, 失效進入錯誤狀態仍然要提供有限的核心功能, 還要提供狀態資訊, 失效的機制或函式必須與其他的機制或函式之間, 分離獨立運作
- 「實作指南」要求密碼學模組必須能被列舉完整的指示器列表, 用以指示每個安全服務自我測試的狀態; 而FIPS 140-2只要求指示操作模式是否為批准核可的模式

2. 密碼學模組的介面 (Cryptographic Module Interfaces)

資料、控制與狀態輸出的介面分離:

- 新增「控制輸出介面」(Control Output Interface) 用於控制指令或控制訊號的輸出

3. 角色、服務與身分認證 (Roles, Services, and Authentication)

新增特定的輸出:

- 自我啟動的密碼學輸出能力 (Self-initiated Cryptographic Output Capability) 保留給重設、重開機、電力循環等使用, 須由組織內的密碼官 (Crypto Officer) 進行設定, 比如開機自動設定 PSec

FIPS 140-3 新增之術語

7. 非侵入式資安 (Non-invasive Security)

Non-invasive attacks attempt to compromise a cryptographic module by acquiring knowledge of the module's Critical Security Parameters without physically modifying or invading the module. Modules may implement various techniques to mitigate against these types of attacks.

「非侵入式攻擊」試圖以非物理性修改或入侵模組之方式，對密碼學模組進行存取，嘗試取得密碼學模組之關鍵安全參數的痕跡知識，得以破壞密碼學模組的機密性。

又稱為「旁路攻擊」(Side-Channel Attack)



The screenshot shows the Wikipedia article for "Van Eck phreaking". The article title is "Van Eck phreaking" and it is categorized under "Article". The article text describes Van Eck phreaking as a form of eavesdropping where special equipment is used to pick up side-band electromagnetic emissions from electronic devices. It mentions that in 1985, Wim van Eck published the first unclassified technical analysis of the security risks of emanations from computer monitors. The article also notes that government researchers were already aware of the danger, as Bell Labs had noted this vulnerability to secure teleprinter communications during World War II. Additionally, the NSA published *Tempest Fundamentals, NSA-82-89, NACSIM 5000, National Security Agency (Classified)* on February 1, 1982. The article concludes by stating that while phreaking is the process of exploiting telephone networks, it is used here because of its connection to eavesdropping. Van Eck phreaking of CRT displays is the process of eavesdropping on the contents of a CRT by detecting its electromagnetic emissions.

https://en.wikipedia.org/wiki/Van_Eck_phreaking

FIPS 140-3 新增之術語

8. 敏感安全參數管理

(Sensitive Security Parameter Management)

- 公開安全參數 (Public Security Parameters, PSP)
要求完整性, 包含公鑰與公開憑證
- 關鍵安全參數 (Critical Security Parameters, CSP)
要求完整性 + 機密性,
包含私鑰、iv值、登入密碼、PIN碼等
- 敏感安全參數 (Sensitive Security Parameter, SSP)
定義為PSP + CSP

9. 自我測試 (Self-Tests)

- 操作前自我測試 (Pre-Operational self-tests)
等同開機自我測試 (Power-On-Self-Test, POST)
- 條件性自我測試 (Conditional self-tests)
依照規格在滿足特定條件時, 進行測試
- 週期性自我測試 (Periodic self-tests)
在安全等級1、2, 等同FIPS 140-2要求
在安全等級3、4, 依照安全政策給定的週期進行自我測試

FIPS 140-3 新增之術語

10. 生命週期確保 (Life-cycle Assurance)

- 由供應者進行之測試(Vendor Testing) 主要包含功能性測試等, 應使用自動化工具進行測試
- 低階測試(Low-level Testing) 安全等級3、4被額外要求, 針對物理連接埠或/與邏輯介面, 進行bitwise等低階測試。測試過程與結果都要進行文件化
- 生命週期結束 (End of life)
安全等級1、2進行敏感資訊消除 (secure sanitization)
安全等級3、4進行損毀 (secure destruction)
- FIPS 140-2要求的進行CVE管理, 被整併到此

FIPS 140-3 安全等級需求 重點摘要

安全等級 1

- ✓ 使用至少一種NIST批准的安全功能或敏感安全參數建立方法
- ✓ 執行於不可修改、受限或可修改的操作環境
- ✓ 硬體式密碼學模組，除了產品級組件 (production-grade components) 之外，不需要物理性資安防護機制；而產品級組件至少要有外封套或可移除的覆蓋物
- ✓ 針對攻擊(包含旁路攻擊)的任何緩解措施，都要文件化

安全等級 2

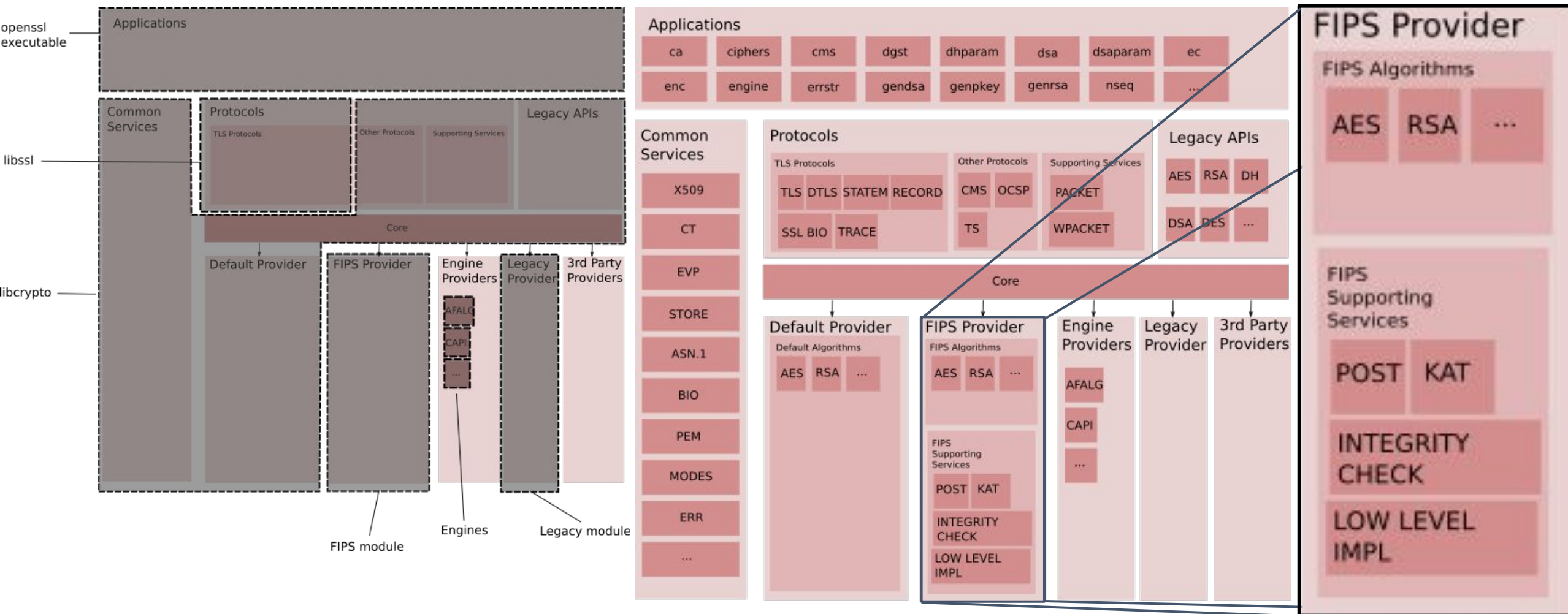
- ✓ 增加防篡改要求，包括在可拆卸的蓋子或門上，使用防篡改塗層(tamper-evident coatings)或封條(seals)或防撬鎖(pick-resistant locks)
- ✓ 基於角色的身份驗證
- ✓ 要在可修改的環境，執行密碼學軟體模組，環境必須實作基於角色的存取控制；或至少透過「存取控制列表」(ACL)酌情處理存取控制；以穩健的機制，定義新群組並且分配限制性權限，與具備將每個使用者帳號分配到多個群組的能力，以防止密碼學軟體模組遭受未經授權的執行、修改和讀取
- ✓ 密碼學模組邊界內的運作環境，只能執行Binary或Bytecode等「可執行碼」，不可以執行源碼、未經過連結器連結的物件碼、或是腳本

FIPS 140-3 安全等級需求 重點摘要

安全等級 3

- ✓ 附加要求對於密碼學模組內所持有的「安全支持提供者」(Security Support Providers)緩解未經授權的存取
- ✓ 物理性資安防護, 防止高機率從通風孔或狹縫探測, 以嘗試偵測密碼學模組並取得模組之回應, 還要防止直接物理性存取、使用或修改密碼學模組
- ✓ 基於身份的身份驗證機制
- ✓ 手動使用「明文密碼服務提供者」(plaintext Cryptographic Service Providers)進行輸入或輸出時, 必須被加密、使用可信通道, 或使用分隔知識的程序
- ✓ 由於超出密碼學模組正常電壓和溫度操作範圍的環境條件, 就有可能會導致安全危害, 需要具有保護密碼學模組免受非常態環境條件破壞的機制
- ✓ 密碼學模組內部實施的任何防禦與緩解非侵入式攻擊的方法, 都必須根據標準文本中, 定義安全等級 3 的衡量指標(metrics)進行測試
- ✓ 提供額外的生命週期確保, 例如: 提供自動化設定值管理、詳細設計、低階測試, 並且使用供應者提供的身份驗證資訊, 進行操作者身份驗證

密碼學模組的設計，以OpenSSL為例



FIPS 140-3 安全等級需求 重點摘要

安全等級 4

- ✓ 對操作者進行多因素身份驗證
- ✓ 密碼學模組包括特定的環保功能，此功能設計能檢測電壓、溫度邊界，以及歸零(zeroize)密碼學模組的「密碼服務提供者」(Cryptographic Service Providers)
- ✓ 密碼學模組內部實施的任何防禦與緩解非侵入式攻擊的方法，都必須根據標準文本中，定義安全等級 4 的衡量指標(metrics)進行測試
- ✓ 輸入到密碼學模組的前提運算條件(pre-conditions)，以及輸出後預期結果為真的後置運算條件(post-conditions)，功能與過程的行為，都要被詳細文件化。

ISO/IEC 19790:2012 安全等級需求 簡表譯文

	安全等級1	安全等級2	安全等級3	安全等級4
密碼學模組規格	密碼學模組的規格、密碼學邊界、批准的密碼學安全函式、與操作上的一般模式或降階模式。密碼學模組的描述要包含所有硬體、軟體與韌體元件。所有的服務都要提供狀態資訊，以顯示某服務正依照准許的規範來利用某個准許的密碼學演算法、安全函式或程序。			
密碼學模組的介面	提供必需的與選擇性的介面。 必須列出所有介面與所有資料輸入輸出的路徑之規格。		使用信任通道。	
角色、服務與身分認證	將必須的與選擇性的角色與服務，進行邏輯分離。	基於角色或身份的操作者認證。	基於身份的操作者認證。	提供多因子身份驗證。
軟韌體資安	使用批准的完整性檢查技術，或基於錯誤追蹤碼(EDC)的完整性測試。 定義軟體、韌體、混合型軟體、混合型韌體等的模組介面。 可執行程式碼。	使用批准的數位簽章，或是基於含金鑰的訊息鑑別碼，以進行完整性測試	基於使用批准的數位簽章，以進行完整性測試。	
作業環境	不可修改，有限修改或可修改。 控制敏感安全參數(SSP)。	可修改。 基於角色的或審慎的存取控制。 稽核機制。		
物理性資安防護	產品級的組件。	保留篡改證據。 不透明覆蓋物或外殼。	篡改偵測與回應覆蓋物或門禁被開啟。 堅固的外殼或塗層。 對於直接探測進行保護。 使用環境失效測試(EFT)或環境失效保護(EFP)。	篡改偵測與回應被拆封。 使用環境失效保護(EFP)。 緩解使用錯誤的注入。
非侵入式資安	模組被設計要對抗與緩解非侵入式的攻擊，規格列於附件F。 將附件F列舉的緩解技術，進行文件化與有效性評估。		緩解測試。	緩解測試。

ISO/IEC 19790:2012 安全等級需求 簡表譯文

		安全等級1	安全等級2	安全等級3	安全等級4
敏感安全參數管理		隨機位元產生器，敏感安全參數產生、建立、輸入和輸出、存儲和歸零。			
		使用批准的方法，進行自動化敏感安全參數傳輸，或使用敏感安全參數協定。			
		手動建立的敏感安全參數，能使用明文進行輸入與輸出。		手動建立的敏感安全參數，輸入與輸出只能使用加密的形式、透過信任通道、或使用分離知識程序。	
自我測試		提供操作前的自我測試：進行軟體/韌體完整性測試、旁路測試、和關鍵功能測試。			
		條件性的自我測試：密碼學演算法測試，成對一致性測試，軟體/韌體載入測試，手動輸入測試，條件性旁路測試和關鍵功能測試。			
生命週期確保	設定值管理	密碼學模組、元件與文件使用的設定值管理系統。每項於整個生命週期，都要被獨特辨識與被追蹤。		自動化設定值管理系統。	
	設計	模組被設計為要允許所有被提供的安全相關測試。			
	有限狀態模型	提供有限狀態模型。			
	研發	源碼、原理圖、或硬體描述語言(HDL)要有註解。	軟體使用高階語言。硬體使用高階描述語言。		當模組元件完成時，預期為真的後置條件，以及輸入到模組元件的先決條件，都要有文件進行註解。
	測試	功能性測試。		低階測試。	
	交付和操作	初始化程序。	交付程序。		使用供應者提供的認證資訊，進行操作者的身分認證。
	指南	提供管理者與非管理者的指南。			
防禦其他攻擊方式		對於緩解攻擊之規格，目前沒有提供可測試的要求。			對於緩解攻擊之規格，有可測試的要求。

5

Summary

由 FIPS 140-2 轉換成 FIPS 140-3 的困難處

參與角色是
密碼學模組
使用者

- 1 密碼學演算法有哪些分類，目的是什麼，要優先搞清楚，建議閱讀密碼學書籍
- 2 密碼學模組規格，介面，以及密碼學模組供應者所提供的使用者手冊都會改版，需要仔細閱讀，積極避免誤用密碼學的輸入參數
- 3 留意哪些密碼學演算法，是密碼學模組在FIPS模式的狀態沒有提供的
- 4 如果版權允許，多參與FIPS 140-3相關的討論
- 5 目前已實作FIPS 140-3的密碼學模組的大廠，都算是早期產品，早期採用者也會面臨一些挑戰

由 FIPS 140-2 轉換成 FIPS 140-3 的困難處

參與角色是
密碼學模組
開發者

- 1 FIPS 140-3其實比FIPS 140-2編排得更有條理
- 2 準備好離散數學和密碼學的知識，才有能力開發「密碼學服務提供者」(Cryptographic Service Provider)，實作核可的密碼學演算法
- 3 每個shall都是一個backlog，解讀backlog需要對密碼學與資訊安全有深入理解
- 4 安全等級3 與 安全等級4 的某些內容還有些模糊
- 5 開發「安全支援提供者」(Security Support Provider)也很重要，比如：已知解答測試(Known Answer Test, KAT)的模組，這些自我測試沒那麼容易開發
- 6 緩解旁路攻擊需要分析電力使用，但SP 800-140F還沒完成，目前尚未清楚對於軟體密碼學模組有怎樣的要求

結語

在SSDLC之中，辨認程式弱點，避免寫出不安全的程式，仍是最關鍵的事務，進行相關的教育訓練，對組織與成員來說都有長遠的幫助

具有一致性、可靠性、受信賴的密碼學模組，是資安關鍵技術之一！
因此密碼學模組需要驗證，國內也有一些廠商有在開發密碼學模組(軟硬體都有)

在SSDLC之中，為了增進資料三態安全性，建議將伺服器的作業系統設定為「啟用FIPS模式」，後端與系統程式將設定值改成FIPS合規的狀態，或是使用FIPS 140-2、140-3合規的函式庫進行密碼學操作，或是直接調用通過FIPS 140-2、140-3認證的密碼學模組所提供的密碼學介面，進行密碼學操作，並且還能夠縮小開發環境與維運環境的差異，使開發人員更能滿足維運人員的需求



THANK YOU!

聯絡我們 contact@onwardsecurity.com



Onward Security



Realize Ultimate Security every step starts with the labs

© 2022 Onward Security Corp. Creative Commons CC0