Appendix Q: Cryptographic Modules

Cryptographic Modules Implemented for Data-in-Transit (DIT)

OSCAL's component model treats independent validation of products and services as if that validation were a separate component. This means when using components with FIPS 140 validated cryptographic modules, there must be two component assemblies:

- **The Validation Definition**: A component that provides details about the validation.
- **The Product Definition**: A component that describes the hardware or software product.

The validation definition is a component that provides details about the independent validation. Its type must have a value of "validation". In the case of FIPS 140 validation, this must include a link field with a rel value set to "validation-details". This link must point to the cryptographic module's entry in the NIST Computer Security Resource Center (CSRC) Cryptographic Module Validation Program Database.

The product definition is a product with a cryptographic module. It must contain all of the typical component information suitable for reference by inventory-items and control statements. It must also include a link field with a rel value set to "validation" and an href value containing a URI fragment. The fragment must start with a hashtag (#) and include the UUID value of the validation component. This links the two together.

Appendix Q <CSO Name> Encryption Implementation Status

Data in Transit (DIT)										
	Source				Destination					
Ref #	Areas of DIT[1]	CMVP #[2]	CM Vendr	Module Name	Areas of DIT	CMVP #[3]	CM Vendor	Module Name	Usage	Notes[4]
1	NGINX Server *<Use Case Example - Please Delete>*	#4271 ☐ Embedded CM ☐ Third-party CM ☒ Uses OS CM ☒ In FIPS Mode ☐ Other	Red Hat, Inc.	RHEL 8 OpenSSL	All Application Servers	#3980 ☐ Embedded CM ☐ Third-party CM ☒ Uses OS CM ☒ In FIPS Mode ☐ Other	Canonical Ltd.	Ubuntu 18.04 OpenSSH Server	Load Balancer TLS to Application Server ☐ TLS 1.1 or earlier ☒ TLS 1.2 ☐ TLS 1.3 ☐ Other _____	
2	All Application Servers *<Use Case Example - Please Delete>*	None ☐ Embedded CM ☐ Third-party CM ☒ Uses OS CM ☐ In FIPS Mode ☐ Other	CentOS 7.9	OpenSSL 1.0.1	PostgreSQL	#3980 ☐ Embedded CM ☐ Third-party CM ☒ Uses OS CM ☒ In FIPS Mode ☐ Other	Canonical Ltd.	Ubuntu 18.04 OpenSSH Server	Application servers to common DB ☐ TLS 1.1 or earlier ☒ TLS 1.2 ☐ TLS 1.3 ☐ Other _____	Plans to move to RHEL 8. See POA&M ID 111.

[1] Each entry should be the component or asset where the FIPS-140 validated cryptographic module is located.
[2] If using cryptography that lacks FIPS validation, state "No FIPS". If unencrypted, state "Unencrypted".
[3] If using cryptography that lacks FIPS validation, state "No FIPS". If unencrypted, state "Unencrypted".
[4] For example, specify if the historical CM is used or the store lacks encryption entirely. Include the related POA&M ID, remediation plans, etc.

Component Representation: Data-In-Transit Example Product with FIPS 140-2 Validation

```
system-security-plan:
  uuid: 11111111-2222-4000-8000-000000000000
  system-implementation:
    components:
    - uuid: 11111111-2222-4000-8000-009000300003
      type: software
      title: OpenSSL
      description: 'Provide a description and any pertinent note regarding the use
        of this CM.'
      props:
      - name: asset-type
        value: cryptographic-module
      - name: version
        value: 3.0.8
      - name: vendor-name
        ns: http://fedramp.gov/ns/oscal
        value: OpenSSL FIPS Provider
      - name: function
        ns: http://fedramp.gov/ns/oscal
        value: data-in-transit
        remarks: Usage statement
      links:
      - href: '#11111111-2222-4000-8000-009001200002'
        rel: validation
        text: A link to the 3rd party validation information related to this cryptographic
          module.
      status:
        state: operational

    - uuid: 11111111-2222-4000-8000-009001200002
      type: validation
      title: OpenSSL FIPS 140-2 Validation
      description: Describe any relevant information regarding this validation of
        the CM.
      props:
      - name: asset-type
        value: cryptographic-module
```

```
      - name: validation-type
        value: fips-140-2
      - name: validation-reference
        value: '4811'
    status:
        state: operational
```

Understanding the Data-in-Transit (DIT) Mapping

When documenting cryptographic protections for data-in-transit, the OSCAL model focuses on the relationship between the specific software provider and its validated state.

- **Software Component & Function:** The first block defines the actual implementation (e.g., **OpenSSL**). The property `name: function` with the value `data-in-transit` explicitly categorizes the module's role. This allows auditors and automated tools to identify which software is responsible for protecting communication channels, such as TLS or SSH connections, across the system boundary.
- **Decoupled Validation Metadata:** Rather than burying version-specific details in a text field, OSCAL uses a `link` to connect the software component to a separate `validation` component. This second component (highlighted by the `validation-reference` value **4811**) points directly to the NIST CMVP certificate.
- **Operational Status:** The `state: operational` field confirms that the module is currently in use within the environment. If a module were undergoing an update or was in a "historical" state, this status could be updated to reflect the current risk posture without needing to rewrite the entire narrative.

By structuring the SSP this way, you ensure that every cryptographic module used for DIT is traceable to a specific FIPS 140-2 or 140-3 certificate, satisfying the requirements for **SC-13 (Cryptographic Protection)** in a machine-verifiable format.

Cryptographic Modules Implemented for Data-at-Rest (DAR)

The approach is the same as in the *cryptographic module data-in-transit* section.

Data at Rest (DAR)							
Ref #	**Areas of DAR[5]**	**CMVP # [6]**	**CM Vendor Name**	**Module Name**	**Usage**	**Encryption Type**	**Notes[7]**
1	PostgreSQL database *<Use Case Example - Please Delete>*	#3980 ☐ Embedded CM ☐ Third-party CM ☒ Uses OS CM ☒ In FIPS Mode ☐ Other _____	Canonical Ltd.	Ubuntu 18.04 OpenSSL Cryptographic Module	Volume encryption	☒ Full disk ☐ File ☐ Record ☐ None ☐ Other _____	
2	App server local storage *<Use Case Example - Please Delete>*	#2931 ☐ Embedded CM ☐ Third-party CM ☒ Uses OS CM ☒ In FIPS Mode ☐ Other _____	Microsoft	Windows Server 2016	OS and application binaries	☐ Full disk ☒ File ☐ Record ☐ None ☐ Other _____	CM is Historical, per NIST CMVP. Plans to move to Windows 2019 upon Active FIPS-140-validation achieved. See POA&M ID 123.
3	S3 buckets *<Use Case Example - Please Delete>*	#4177 ☒ Embedded CM ☐ Third-party CM ☐ Uses OS CM ☒ In FIPS Mode ☐ Other _____	AWS	Key Management Service (KMS) HSM	Server-side encryption with KMS keys (SSE-KMS) used to encrypt bucket	☒ Full disk ☐ File ☐ Record ☐ None ☐ Other _____	

[5] Each entry should be the component or asset where the FIPS-140 validated cryptographic module is located.
[6] If using cryptography that lacks FIPS validation, state "No FIPS". If unencrypted, state "Unencrypted".
[7] For example, specify if the historical CM is used or the store lacks encryption entirely. Include the related POA&M ID, remediation plans, etc.

Component Representation: Data-At=Rest Example Product with FIPS 140-2 Validation

```
system-security-plan:
  uuid: 11111111-2222-4000-8000-000000000000
  system-implementation:
    components:
    - uuid: 11111111-2222-4000-8000-009000300012
      type: software
      title: Database Row Encryption Module
      description: Briefly describe the cryptographic module.
      props:
      - name: asset-type
        value: cryptographic-module
      - name: version
        value: 1.2.3
      - name: vendor-name
        ns: http://fedramp.gov/ns/oscal
        value: Databases-R-Us
      - name: function
```

```
      ns: http://fedramp.gov/ns/oscal
      value: data-at-rest
      remarks: Used to encrypt and decrypt rows in the database.
    status:
      state: operational

  - uuid: 11111111-2222-4000-8000-009001200001
    type: validation
    title: Database Row Encryption Module (DREM)
    description: Briefly describe the cryptographic module.
    props:
    - name: asset-type
      value: cryptographic-module
    - name: validation-type
      value: fips-140-2
    - name: validation-reference
      value: '0000'
    status:
      state: operational
```

Understanding the Data-at-Rest (DAR) Mapping

In the OSCAL representation of data-at-rest protections, the focus shifts from communication protocols to the specific encryption mechanisms securing stored information.

- **Defining the Storage Function:** The property `name: function` with the value `data-at-rest` explicitly categorizes the module's role. The accompanying `remarks` field—such as "Used to encrypt and decrypt rows in the database"—provides the necessary context for human reviewers to understand the scope of the encryption (e.g., full-disk vs. application-layer encryption).
- **Asset Categorization:** By using the `asset-type: cryptographic-module` property, the component is tagged for automated compliance auditing. This allows the system to verify that every component handling sensitive federal data is linked to a valid cryptographic implementation, directly supporting **SC-28 (Protection of Information at Rest)**.
- **Validation Linkage:** Similar to the data-in-transit model, the software component is linked to a `validation` component that holds the NIST CMVP metadata. The `validation-reference` (e.g., **0000**) acts as the source of truth for the FIPS 140-2 or 140-3 certificate number, ensuring that the module meets the mandatory federal security standards for data storage.

By organizing DAR in this manner, the SSP provides a granular inventory of encryption at every layer of the technology stack—from the database row level up to the storage volume—while maintaining a clear audit trail to the validated cryptographic provider.

> **NOTE:**
>
> While the examples show FIPS 140-2, the same OSCAL structure applies to FIPS 140-3. Simply update the `validation-type` property to reflect the current standard.

Revision #12
Created 2026-02-11 23:03:59 UTC by Brian Ruf
Updated 2026-04-09 00:53:29 UTC by Brian Ruf