*The Johns Hopkins University Applied Physics Laboratory*

*Sarah Heiner*

[*Sarah.Heiner@jhuapl.edu*](mailto:Sarah.Heiner@jhuapl.edu)

*June 2022*

*BPSec Python Test Suite*

BPSec Test Suite for ION Open Source (ION-IOS) 4.1.2

**Table of Contents**

[1 Overview 2](#_Toc105500260)

[2 Configuring ION for BPSec Testing 2](#_Toc105500261)

[3 Running the Test Suite 3](#_Toc105500262)

[3.1 Test Cases 3](#_Toc105500263)

[3.2 Test Macros 3](#_Toc105500264)

[3.3 Test Output 4](#_Toc105500265)

[3.3.1 Default Output 4](#_Toc105500266)

[3.3.2 Detailed Output 4](#_Toc105500267)

[3.3.3 Verbose Output 5](#_Toc105500268)

[3.4 Test Debugging 6](#_Toc105500269)

[3.5 Test Help 7](#_Toc105500270)

[4 BPSec Test Case Definitions 7](#_Toc105500271)

[5 Test Case Current Results: IOS 4.1.2 17](#_Toc105500272)

# Overview

This Python test suite has been written to test the ION Open Source (IOS) BPSec release candidate, the IETF default security contexts, and the BPSec ION Policy Software (BIPS) that supports them.

The included test cases address both bib-integrity and bcb-confidentiality security operations. The BIB-HMAC-SHA default security context is used for bib-integrity operations, and BCB-AES-GCM is used for bcb-confidentiality operations.

Providing backward compatibility with previous ION capabilities, the security context *ION Test Security Context* (ITSC) is included in addition to the BPSec Default Security Contexts from RFC 9173. The ITSC can be used for both bib-integrity and bcb-confidentiality security operations and supports the BPSec capabilities provided with previous ION releases:

bib-integrity operations on the Primary Block.

bib-integrity operations on the Payload Block.

bcb-confidentiality operations on the Payload Block.

The rest of this document is laid out as follows: Section 2 Configuring ION for BPSec Testing reviews the system requirements and build options necessary to run this test suite. Section 3 Running the Test Suite details the command line options this test suite provides, allowing for customized use of the test suite. Section 4 BPSec Test Case Definitions describes each of the test cases in detail, and Section 5 Test Case Current Results: IOS 4.1.2 provides an summary of current testing results.

# Configuring ION for BPSec Testing

This Python test suite requires the installation of **python3** to run.

The test suite verifies the success of each test using test events, which are statements found in the ION log for each node. In order to use this test suite, the logging of test events must be turned on. This step must be taken before building ION.

To enable test-level logging:

1. Enable test points in bib.h. Navigate to /bpv7/library/ext/bpsec/bib.h and modify the **#define BIB\_TEST\_LOGGING** to be set to **1.**
2. Enable test points in bcb.h. Navigate to /bpv7/library/ext/bpsec/bcb.h and modify the **#define BCB\_TEST\_LOGGING** to be set to **1.**
3. Build ION.
4. Run the test suite. Further instructions can be found in Section 3 Running the Test Suite.

# Running the Test Suite

The python test suite should be run from the directory that dotest.py is located in:

**/tests/bpsec/python\_tests**

To run the python test suite, use the following command:

**python3 dotest.py [-h] [--test\_case TEST\_CASE [TEST\_CASE ...]]**

**[--test\_macro TEST\_MACRO] [--output default | detailed | verbose] [--debug]**

## Test Cases

To execute one or more tests, the user has the option to identify them by test case number or grouping macros.

If identifying tests to run by their test ID (test case number), include the **--test\_case** argument, followed by the test case(s) to run, separated by spaces.

To run test case 1, use the following command:

**python3 dotest.py --test\_case 1**

To run test cases 1, 2, and 5, use the following command:

**python3 dotest.py --test\_case 1 2 5**

## Test Macros

To run one of the supported groups of tests, provide one of the following test macros rather than a single test case number:

* **all** – Run all of the BPSec test cases.
* **bib** – Run all of the BPSec test cases that use the bib-integrity service.
* **bcb** – Run all of the BPSec test cases that use the bcb-confidentiality service.
* **payload** – Run all of the BPSec test cases that specify the Payload Block as a security target of a security operation.
* **primary** – Run all of the BPSec test cases that specify the Primary Block as a security target of a security operation.

To run all of the BPSec test cases provided with this test suite, use the following command:

**python3 dotest.py --test\_macro all**

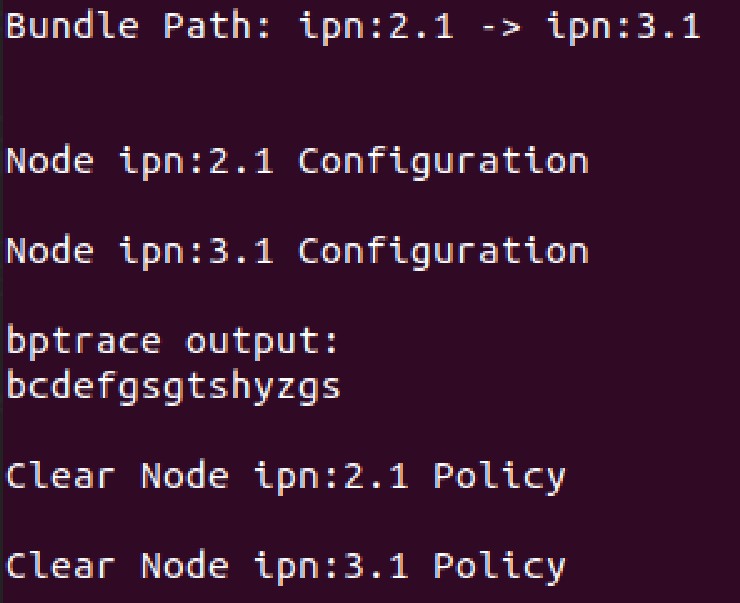
## Test Output

Test output can be configured depending on the level of detail a user prefers. The optional **--output** argument is automatically set to **default** and only needs to be specified if the user would prefer the **detailed** or **verbose** output to be displayed.

### Default Output

Running tests with **--output** set to **default** will display only the bundle path, the nodes where security policy is configured by the test case, and the output from the bptrace command (if executed by the test case).

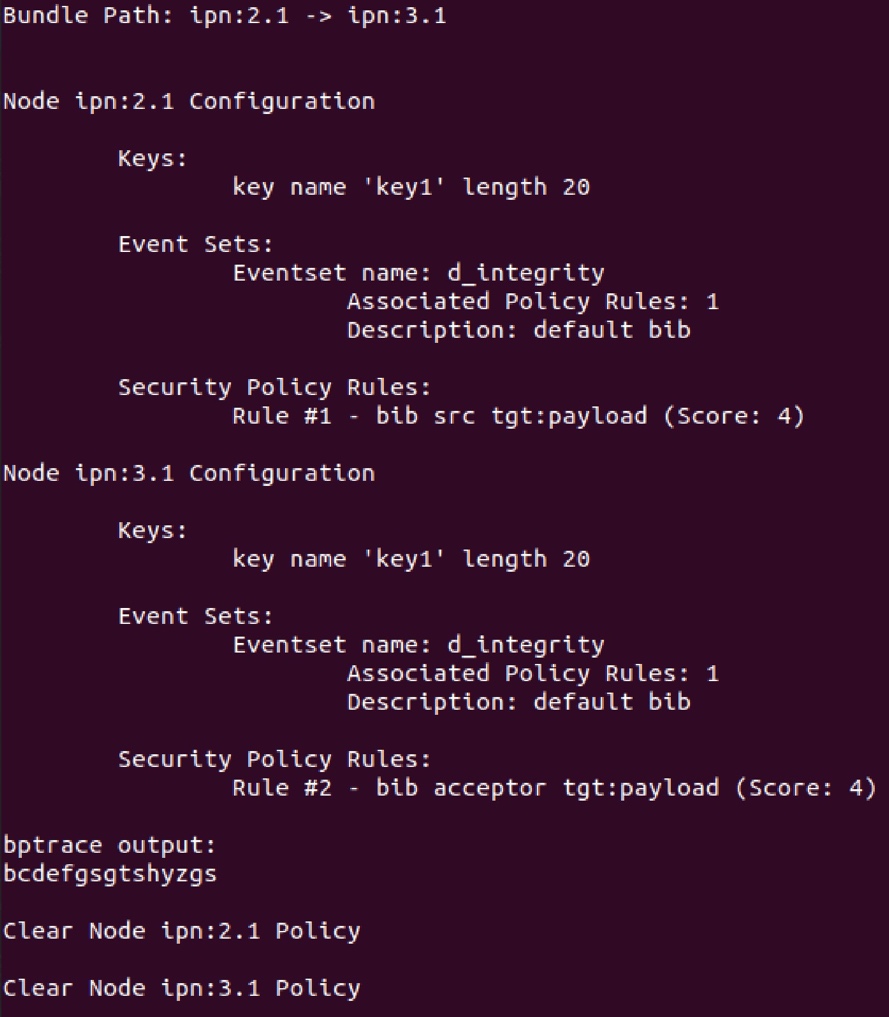
Below is sample output from Test Case 1 run with test output set to default.



### Detailed Output

Running tests with **--output** set to **detailed** will display the bundle path, output from the bptrace command (if executed by the test case), and an overview of the security policy configuration at each node. This overview includes a list of all keys present, event sets and their descriptions, as well as a summary of any security policy rules defined at that node.

Below is sample output from Test Case 1 run with test output set to detailed.

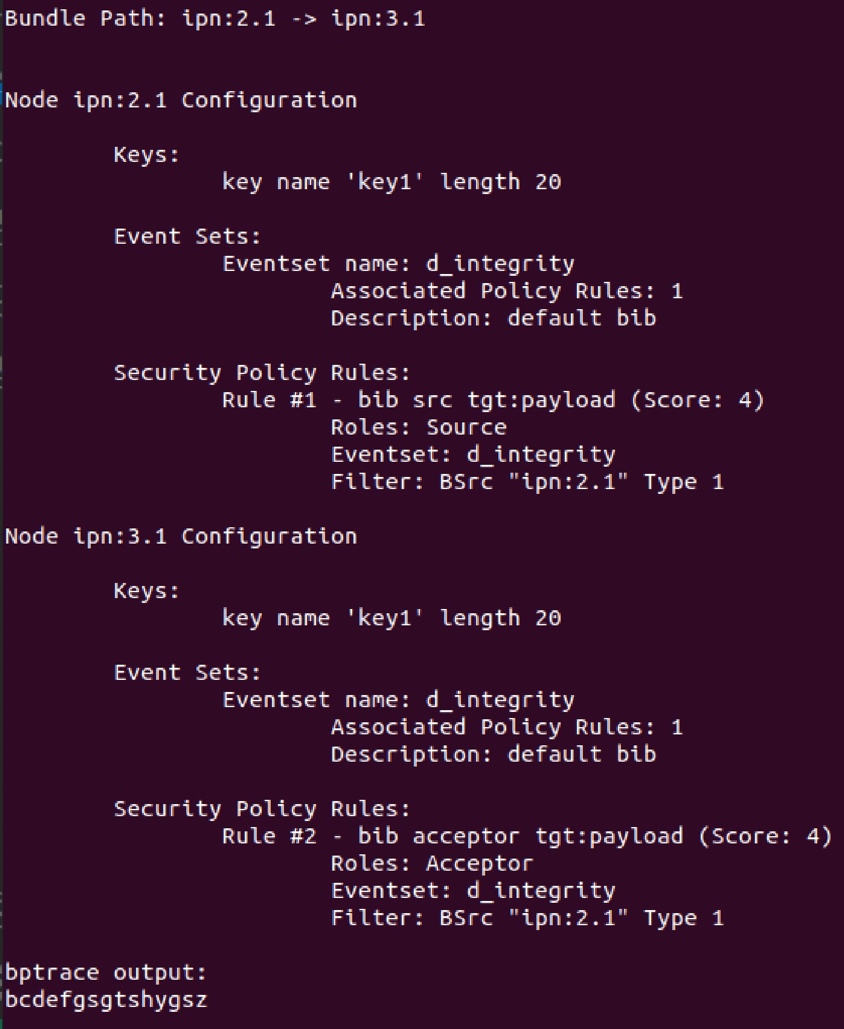


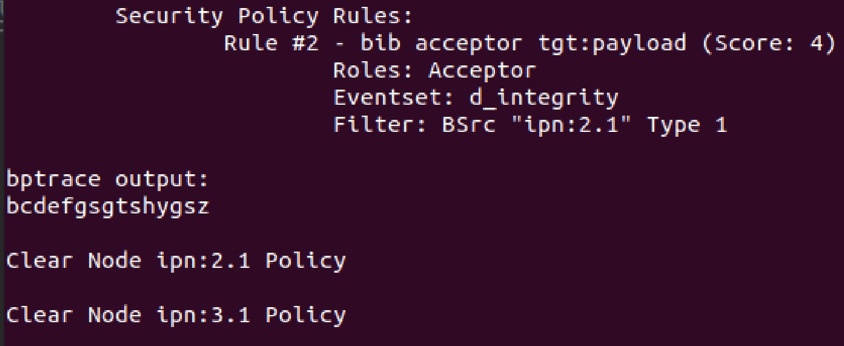
### Verbose Output

Running tests with **--output** set to **verbose** will displaywill display the bundle path, output from the bptrace command (if executed by the test case), and the security policy configuration at each node. The security policy information includes:

1. A list of all keys present
2. All Event Sets, their descriptions, and the number of security policy rules they are associated with.
3. The security policy rules defined at that node, with the security service, security target block, security role, event set, and filter criteria identified for each rule.

Below is sample output from Test Case 1 run with test output set to verbose.

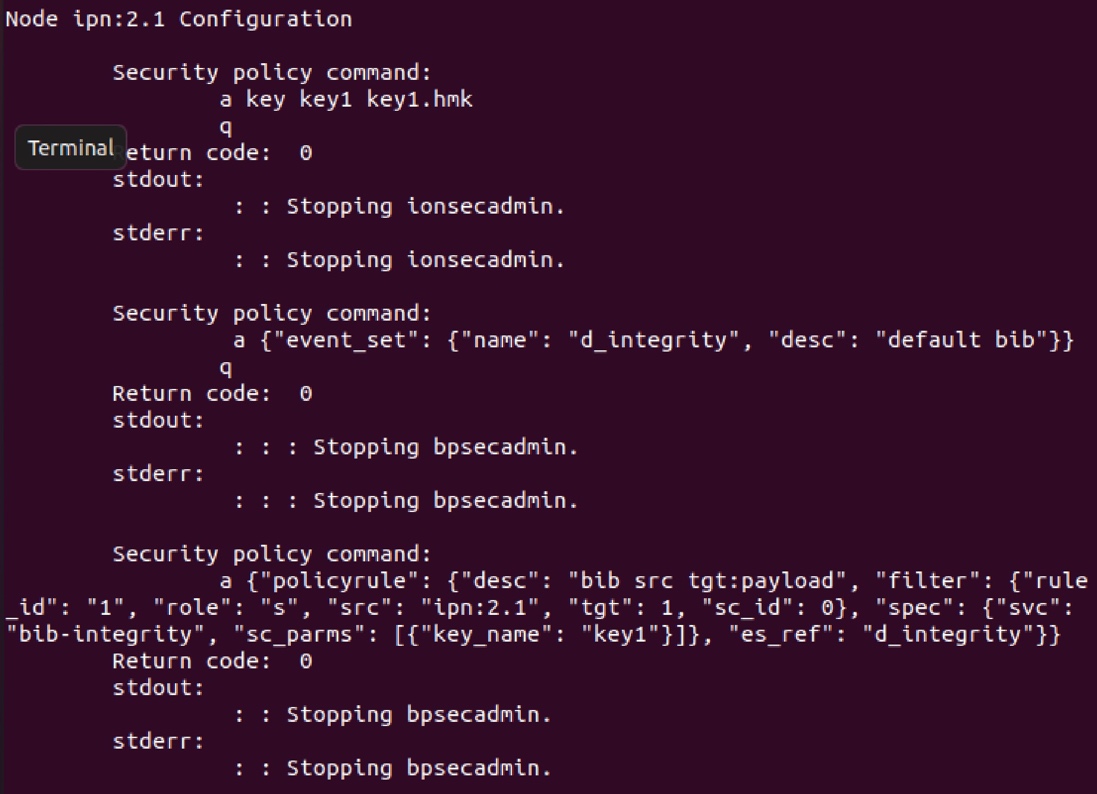




## Test Debugging

If the BPSec python test suite is being used by a developer, they may wish to turn on debugging. To enable debugging, include the **--debug** argument when running the test suite. With debugging enabled, the test suite displays the security policy command(s) issued by the test cases at each node, as well as the return code and stdout and stderr output from the utility used for security configuration – either ionsecadmin or bpsecadmin.

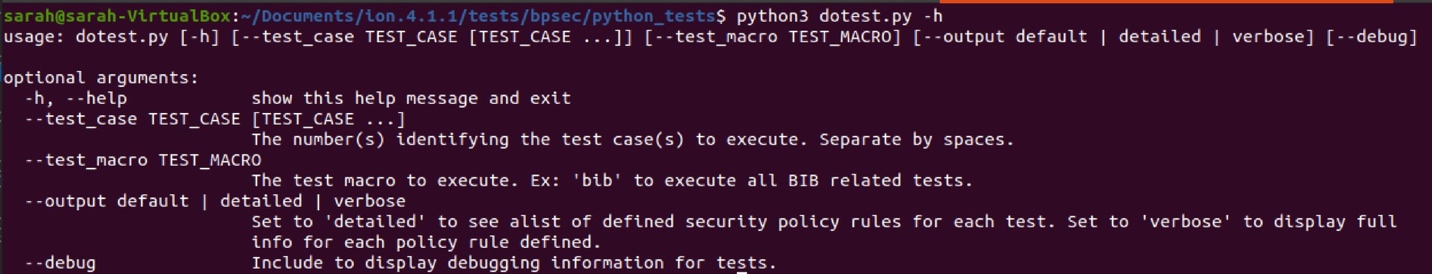
Below is sample output from Test Case 1 run with the debug option enabled.



## Test Help

For additional help with using the BPSec python test, add the **-h** argument. This will call up a help message as a reminder of command line options for the test suite.

Below is the expected output from running **python3 dotest.py -h**



# BPSec Test Case Definitions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Purpose | Description | Security Service(s) | Bundle Path |
| 1 | Test Payload block integrity. | Add a BIB targeting a Payload Block at Security Source ipn:2.1 and process that BIB at Security Acceptor ipn:3.1. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 2 | Test Payload block confidentiality. | Add a BCB targeting a Payload Block at Security Source ipn:2.1 and process that BCB at Security Acceptor ipn:3.1. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 3 | Test Primary block integrity. | Add a BIB targeting a Primary Block at Security Source ipn:2.1 and process that BIB at Security Acceptor ipn:3.1. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 4 | Check that a BCB is not permitted to target a Primary block. | Attempt to add a BCB targeting a Primary Block, which is not permitted by BPSec, at Security Source ipn:2.1 and process that BCB at Security Acceptor ipn:3.1. A BCB should not be added to this bundle. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 5 | BIB behavior at a Security Source, Security Verifier, and Security Acceptor. | Add a BIB targeting a Payload Block at Security Source ipn:2.1, transmit to the Security Verifier  ipn:3.1 to check integrity and process that BIB at Security Acceptor ipn:4.1. | bib-integrity | ipn:2.1 ->  ipn:3.1 ->  ipn:4.1 |
| 6 | BCB behavior at a Security Source, Security Verifier, and Security Acceptor. | Add a BCB targeting a Payload Block at Security Source ipn:2.1 and transmit to the Security Verifier  ipn:3.1. Note that verification of a BCB currently does nothing in ION, acting simply as a pass-through. Process that BCB at Security Acceptor ipn:4.1. | bcb-confidentiality | ipn:2.1 ->  ipn:3.1 ->  ipn:4.1 |
| 7 | Check that duplicate security operations – in this case, a BIB on the Payload block – are prevented.  Ensure that rule prioritization works. | Create duplicate security policy rules at a Security Source requiring a BIB targeting the Payload block. A single BIB should be added to the bundle, indicating that rule prioritization behaves as expected. Configure the Security Acceptor with duplicate rules as well, expecting to see that the single BIB is processed. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 8 | Check that duplicate security operations – in this case, a BCB on the Payload block – are prevented.  Ensure that rule prioritization works. | Create duplicate security policy rules at a Security Source requiring a BCB targeting the Payload block. A single BCB should be added to the bundle, indicating that rule prioritization behaves as expected. Configure the Security Acceptor with duplicate rules as well, expecting to see that the single BCB is processed. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 9 | Check that a misconfiguration between the Security Source and Acceptor is caught.  Misconfiguration: Use a different integrity SHA variant for the same security context at the Security Source and Security Acceptor. | Create a security policy rule for a Security Source at ipn:2.1 for a BIB on the Payload block using security context BIB-HMAC-SHA2 with SHA variant 5 – HMAC 256/256. Create a misconfigured security policy rule for the Security Acceptor at ipn:3.1 using BIB-HMAC-SHA2 with SHA variant 6 – HMAC 384/384 and detect this misconfiguration. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 10 | Check that a misconfiguration between the Security Source and Acceptor is caught.  Misconfiguration: Use a different AES variant for the same confidentiality security context at the Security Source and Security Acceptor. | Create a security policy rule for a Security Source at ipn:2.1 for a BCB on the Payload block using security context BCB-AES-GCM with AES variant 1 – A128GCM. Create a misconfigured security policy rule for the Security Acceptor at ipn:3.1 using BCB-AES-GCM with AES variant 3 – A256GCM and detect this misconfiguration. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 11 | Check that a Security Acceptor can identify a missing BIB.  Security operation event associated: sop\_missing\_at\_acceptor | Create a Security Acceptor rule at ipn:3.1 requiring a BIB on the Primary Block. Omit security source rule at ipn:2.1 and expect to see the sop\_missing\_at\_acceptor test event.  Intentional misconfiguration of security policy. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 12 | Check that a Security Acceptor can identify a missing BCB.  Security operation event associated: sop\_missing\_at\_acceptor | Create a Security Acceptor rule at ipn:3.1 requiring a BCB on the Payload Block. Omit security source rule at ipn:2.1 and expect to see the sop\_missing\_at\_acceptor test event.  Intentional misconfiguration of security policy. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 13 | Check that a Security Acceptor can identify a corrupted BIB target.  Security operation event associated: sop\_corrupted\_at\_acceptor. | Transmit a bundle with a BIB targeting a payload block that is corrupted and cannot be processed and integrity verified at the Security Acceptor. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 14 | Check that a Security Acceptor can identify a corrupted BCB target.  Security operation event associated: sop\_corrupted\_at\_acceptor. | Transmit a bundle with a BCB targeting a payload block that is corrupted and cannot be processed and decrypted at the Security Acceptor. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 15 | Check that a Security Verifier can identify a corrupted BIB target.  Security operation event associated: sop\_corrupted\_at\_verifier. | Transmit a bundle with a BIB targeting a payload block that is corrupted and cannot be processed and integrity verified at the Security Verifier. | bib-integrity | ipn:2.1 -> ipn:3.1 ->  ipn:4.1 |
| 16 | Check that a Security Verifier can identify a corrupted BCB target.  Security operation event associated: sop\_corrupted\_at\_verifier. | Transmit a bundle with a BCB targeting a payload block that is corrupted and cannot be processed (its AAD as a Security Verifier does not perform decryption) at the Security Verifier. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 ->  ipn:4.1 |
| 17 | Check that a Security Acceptor can identify a misconfigured BIB due to use of mismatched keys.  Security operation event associated: sop\_misconfigured\_at\_acceptor. | Create a Security Source rule at ipn:2.1 using key1 (key1.hmk) for a BIB on the Payload block and intentionally misconfigure the Security Acceptor to use key2 (key2.hmk).  Check that the Security Acceptor acknowledges the BIB misconfiguration. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 18 | Check that a Security Acceptor can identify a misconfigured BCB due to use of mismatched keys.  Security operation event associated: sop\_misconfigured\_at\_acceptor. | Create a Security Source rule at ipn:2.1 using bcbkey (bcbkey.hmk) for a BCB on the Payload block and intentionally misconfigure the Security Acceptor to use bcbkey2 (bcbkey2.hmk).  Check that the Security Acceptor acknowledges the BIB misconfiguration. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 19 | Check that a Security Verifier can identify a misconfigured BIB.  Security operation event associated: sop\_misconfigured\_at\_verifier. | Create a Security Source rule at ipn:2.1 using key1 (key1.hmk) for a BIB on the Payload block and intentionally misconfigure the Security Verifier to use key2 (key2.hmk). Check that the Security Verifier acknowledges the BIB misconfiguration. | bib-integrity | ipn:2.1 -> ipn:3.1 -> ipn:4.1 |
| 20 | Check that a Security Verifier can identify a misconfigured BCB.  Security operation event associated: sop\_misconfigured\_at\_verifier. | Create a Security Source rule at ipn:2.1 using key1 (key1.hmk) for a BCB on the Payload block and intentionally misconfigure the Security Verifier to use key2 (key2.hmk) to check the BCB’s AAD. Check that the Security Verifier acknowledges the BCB misconfiguration. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 -> ipn:4.1 |
| 21 | Generate an integrity signature for a large payload (1MB). | Add a BIB to the bundle targeting a large Payload Block (1MB) and process that BIB at the security acceptor, ensuring that the entire payload reaches its destination. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 22 | Check that a misconfiguration between the Security Source and Security Acceptor is caught.  Misconfiguration: Attempt to use a confidentiality security context when requiring a bib-integrity service. | Create a security policy rule for a Security Source at ipn:2.1 for a BIB on the Payload block using security context BIB-HMAC-SHA2. Create a misconfigured security policy rule for the Security Acceptor at ipn:3.1 using the wrong security context BCB-AES-GCM. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 23 | Check that a misconfiguration between the Security Source and Security Acceptor is caught.  Misconfiguration: Attempt to use an integrity security context when requiring a bcb-confidentiality service. | Create a security policy rule for a Security Source at ipn:2.1 for a BCB on the Payload block using security context BCB-AES-GCM. Create a misconfigured security policy rule for the Security Acceptor at ipn:3.1 using the wrong security context BIB-HMAC-SHA2. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 24 | Check that the BPSec implementation does not permit a BCB to target another BCB. | Create a security source rule at ipn:2.1 requiring a BCB on the Payload Block. Add a second security source rule at the same node requiring a BCB targeting the first one created.  Expect to see a BCB on the Payload Block only, as a BCB targeting another BCB is prohibited by BPSec. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 25 | Check that a BIB and BCB are permitted to share a security target. | Create a security source rule at ipn:2.1 requiring a BCB on the Payload Block. Add a second security source rule at the same node requiring a BIB targeting the Payload Block as well. Add security acceptor rules at ipn:3.1 for both the BIB and BCB. | bib- integrity  bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 26 | Check that a BIB is not added at a waypoint node if a BCB already exists with the same target block. | Create a bundle with a BCB targeting the Payload Block. Attempt to add a BIB at a waypoint node targeting the Payload Block. The BIB should not be added. Process the BCB at the security acceptor.  Create a security source rule at ipn:2.1 requiring a BCB on the Payload Block. Create a security source rule at waypoint node ipn:3.1 requiring a BIB targeting the Payload Block. Note that BPSec prohibits a BIB with a target block that is already encrypted by a BCB. The BIB should NOT be added to the bundle. Add a security acceptor rule at ipn:4.1 for the BCB. | bib- integrity  bcb-confidentiality | ipn:2.1 ->  ipn:3.1 ->  ipn:4.1 |
| 27 | Check that the BPSec implementation does not permit a BIB to target another BIB. | Create a security source rule at ipn:2.1 requiring a BIB on the Payload Block. Add a second security source rule at the same node requiring a BIB targeting the first one created.  Expect to see a BIB on the Payload Block only, as a BIB targeting any security block is prohibited by BPSec. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 28 | Check that a BCB is not permitted to target a BIB it does not share a security target with. | Create a bundle with a BIB targeting the Payload Block. Designate a waypoint node as the security source for a BCB whose target is that BIB.  This behavior is prohibited by BPSec as the BCB does not share a security target with the BIB.  Check that only the BIB is added to the bundle and that it can be processed at its security acceptor. | bib-integrity  bcb-confidentiality | ipn:2.1 ->  ipn:3.1 ->  ipn:4.1 |
| 29 | Check that a BCB can encrypt a BIB if the two security blocks share a security target. | Configure node ipn:2.1 to be a security source for both a BIB and BCB. Both of these security blocks share the same target: the Payload Block.  A BCB and BIB may share a target if the BCB encrypts the BIB to remain compliant with BPSec.  Node ipn:3.1 serves as the security acceptor for both security services. | bib-integrity  bcb-confidentiality | ipn:2.1 ->  ipn:3.1 |
| 30 | Check that a BCB can be added to the bundle at a waypoint node if it shares all of an existing BIB’s targets.  Target multiplicity and security block interactions. | Configure node ipn:2.1 to be the Security Source for a BIB targeting an extension block and the Payload Block.  Configure waypoint node ipn:3.1 to be the Security Source for a BCB targeting that same extension block and the Payload Block.  Node ipn:4.1 is configured to be the Security Acceptor for all four security operations. | bib-integrity  bcb-confidentiality | ipn:2.1 ->  ipn:3.1 ->  ipn:4.1 |
| 31 | Check that a BCB can be added to the bundle at a waypoint node if it shares some of an existing BIB’s targets.  Target multiplicity and security block interactions. | Configure node ipn:2.1 to be the Security Source for a BIB targeting the Primary Block and the Payload Block.  Configure waypoint node ipn:3.1 to be the Security Source for a BCB targeting the Payload Block.  Node ipn:4.1 is configured to be the Security Acceptor for all three security operations. | bib-integrity  bcb-confidentiality | ipn:2.1 ->  ipn:3.1 ->  ipn:4.1 |
| 32 | Check that a BCB can be added to the bundle at a waypoint node if it shares some of an existing BIB’s targets.  Target multiplicity and security block interactions. | Configure node ipn:2.1 to be the Security Source for a BIB targeting the Payload Block.  Configure waypoint node ipn:3.1 to be the Security Source for a BCB targeting the Payload Block and a non-security extension block.  Node ipn:4.1 is configured to be the Security Acceptor for all three security operations. | bib-integrity  bcb-confidentiality | ipn:2.1 ->  ipn:3.1 ->  ipn:4.1 |
| 33 | BIB target multiplicity  Check that a BIB can have multiple targets of different block types. | Configure node ipn:2.1 to be a security source for a BIB targeting both the Payload and Primary blocks.  Node ipn:3.1 serves as the security acceptor for both security operations. | bib-integrity | ipn:2.1 ->  ipn:3.1 |
| 34 | BCB target multiplicity  Check that a BCB can have multiple targets of different block types. | Configure node ipn:2.1 to be a security source for a BCB targeting both the Payload and an extension block.  Node ipn:3.1 serves as the security acceptor for both security operations. | bcb-confidentiality | ipn:2.1 ->  ipn:3.1 |
| 35 | BIB target multiplicity  Check that a BIB can have multiple targets of the same block type. | Configure node ipn:2.1 to be a security source for a BIB targeting two extension blocks of the same type.  Node ipn:3.1 serves as the security acceptor for both security operations. | bib-integrity | ipn:2.1 ->  ipn:3.1 |
| 36 | BCB target multiplicity  Check that a BCB can have multiple targets of the same block type. | Configure node ipn:2.1 to be a security source for a BCB targeting two extension blocks of the same type.  Node ipn:3.1 serves as the security acceptor for both security operations. | bcb-confidentiality | ipn:2.1 ->  ipn:3.1 |
| 37 | Multi bundle test | Ensure that the test framework is tracking security operation events on a per-bundle basis.  Configure nodes ipn:2.1 and ipn:3.1 to be security sources for a BIB targeting the payload block.  Configure node ipn:4.1 to serve as the security acceptor for all BIBs regardless of bundle source.  Create two bundles, one at ipn:2.1 and one at ipn:3.1 both with destination ipn:4.1 and ensure that BIB processing events are handled separately by the test framework. | bib-integrity | Bundle 1:  Ipn:2.1 -> ipn:4s.1  Bundle 2: ipn:3.1 -> ipn:4.1 |
| 38 | Check that a Security Verifier can identify a missing BIB.  Security operation event associated: sop\_missing\_at\_verifier. | Create a Security Verifier rule at ipn:3.1 requiring a BIB on the Primary Block. Omit security source rule at ipn:2.1 and expect to see the sop\_missing\_at\_verifier test event.  Intentional misconfiguration of security policy. | bib-integrity | ipn:2.1 -> ipn:3.1 ->  ipn:4.1 |
| 39 | Check that a Security Verifier can identify a missing BCB.  Security operation event associated: sop\_missing\_at\_verifier. | Create a Security Verifier rule at ipn:3.1 requiring a BCB on the Payload Block. Omit security source rule at ipn:2.1 and expect to see the sop\_missing\_at\_verifier test event.  Intentional misconfiguration of security policy. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 ->  ipn:4.1 |
| 40 | Exercise the bpsecadmin ‘find’ command. | Testing the bpsecadmin find command, performing searches with find types set to 'all' or 'best'. | N/A | ipn:2.1 (no transmission necessary). |
| 41 | Check that the security context ID field can accept an integer or string, and that invalid security context IDs are identified. | Create policy rules using both string and integer values to represent the associated security context. This test checks that   1. Only policy rules with supported security context IDs are created. 2. The sc\_id field can accept string or integer security context identifiers. | N/A | ipn:2.1 (no transmission necessary). |
| 42 | BIB targeting an Extension Block. | Add a BIB targeting the Bundle Age Block at Security Source ipn:2.1 and process that BIB at Security Acceptor ipn:3.1. | bib-integrity | ipn:2.1 -> ipn:3.1 |
| 43 | BCB targeting an Extension Block. | Add a BCB targeting the Bundle Age Block at Security Source ipn:2.1 and process that BCB at Security Acceptor ipn:3.1. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |
| 44 | Encrypt a large payload (1MB). | Add a BCB to the bundle targeting a large Payload Block (1MB) and process that BCB at the security acceptor, ensuring that the entire payload reaches its destination. | bcb-confidentiality | ipn:2.1 -> ipn:3.1 |

# Test Case Current Results: IOS 4.1.2

|  |  |  |
| --- | --- | --- |
| Test ID | Purpose | Test Status |
| 1 | Test Payload block integrity. | Passed. |
| 2 | Test Payload block confidentiality. | Passed. |
| 3 | Test Primary block integrity. | Passed. |
| 4 | Check that a BCB is not permitted to target a Primary block. | Passed. |
| 5 | BIB behavior at a Security Source, Security Verifier, and Security Acceptor. | Passed. |
| 6 | BCB behavior at a Security Source, Security Verifier, and Security Acceptor. | Passed. |
| 7 | Check that duplicate security operations – in this case, a BIB on the Payload block – are prevented.  Ensure that rule prioritization works. | Passed. |
| 8 | Check that duplicate security operations – in this case, a BCB on the Payload block – are prevented.  Ensure that rule prioritization works. | Passed. |
| 9 | Check that a misconfiguration between the Security Source and Acceptor is caught.  Misconfiguration: Use a different integrity security context at the Security Source and Security Acceptor. | To be used for INB testing. |
| 10 | Check that a misconfiguration between the Security Source and Acceptor is caught.  Misconfiguration: Use a different confidentiality security context at the Security Source and Security Acceptor. | To be used for INB testing. |
| 11 | Check that a Security Acceptor can identify a missing BIB.  Security operation event associated: sop\_missing\_at\_acceptor | Not supported.  Identification of missing security operations will be implemented in the ION 4.2 release. |
| 12 | Check that a Security Acceptor can identify a missing BCB.  Security operation event associated: sop\_missing\_at\_acceptor | Not supported.  Identification of missing security operations will be implemented in the ION 4.2 release. |
| 13 | Check that a Security Acceptor can identify a corrupted BIB target.  Security operation event associated: sop\_corrupted\_at\_acceptor. | Not implemented.  To be completed for the ION 4.2 release. |
| 14 | Check that a Security Acceptor can identify a corrupted BCB target.  Security operation event associated: sop\_corrupted\_at\_acceptor. | Not implemented.  To be completed for the ION 4.2 release. |
| 15 | Check that a Security Verifier can identify a corrupted BIB target.  Security operation event associated: sop\_corrupted\_at\_verifier. | Not implemented.  To be completed for the ION 4.2 release. |
| 16 | Check that a Security Verifier can identify a corrupted BCB target.  Security operation event associated: sop\_corrupted\_at\_verifier. | Not implemented.  To be completed for the ION 4.2 release. |
| 17 | Check that a Security Acceptor can identify a misconfigured BIB due to use of mismatched keys.  Security operation event associated: sop\_misconfigured\_at\_acceptor. | To be used for INB testing. |
| 18 | Check that a Security Acceptor can identify a misconfigured BCB due to use of mismatched keys.  Security operation event associated: sop\_misconfigured\_at\_acceptor. | To be used for INB testing. |
| 19 | Check that a Security Verifier can identify a misconfigured BIB.  Security operation event associated: sop\_misconfigured\_at\_verifier. | To be used for INB testing. |
| 20 | Check that a Security Verifier can identify a misconfigured BCB.  Security operation event associated: sop\_misconfigured\_at\_verifier. | To be used for INB testing.  Note: The Security Verifier role for bcb-confidentiality operations will be implemented in the ION 4.2 release. |
| 21 | BIB targeting a large payload (1MB). | Passed. |
| 22 | Check that a misconfiguration between the Security Source and Security Acceptor is caught.  Misconfiguration: Attempt to use a confidentiality security context when requiring a bib-integrity service. | To be used for INB testing. |
| 23 | Check that a misconfiguration between the Security Source and Security Acceptor is caught.  Misconfiguration: Attempt to use an integrity security context when requiring a bcb-confidentiality service. | To be used for INB testing. |
| 24 | Check that the BPSec implementation does not permit a BCB to target another BCB. | Not supported.  This block interaction will be implemented in the ION 4.2 release. |
| 25 | Check that a BIB and BCB are permitted to share a security target. | Not supported.  This block interaction will be implemented in the ION 4.2 release. |
| 26 | Check that a BIB is not added at a waypoint node if a BCB already exists with the same target block. | Not supported.  This block interaction will be implemented in the ION 4.2 release. |
| 27 | Check that the BPSec implementation does not permit a BIB to target another BIB. | Not supported.  This block interaction will be implemented in the ION 4.2 release. |
| 28 | Check that a BCB is not permitted to target a BIB it does not share a security target with. | Not supported.  This block interaction will be implemented in the ION 4.2 release. |
| 29 | Check that a BCB can encrypt a BIB if the two security blocks share a security target. | Not supported.  This block interaction will be implemented in the ION 4.2 release. |
| 30 | Check that a BCB can be added to the bundle at a waypoint node if it shares all of an existing BIB’s targets.  Target multiplicity and security block interactions. | Not supported.  This block interaction will be implemented in the ION 4.2 release. |
| 31 | Check that a BCB can be added to the bundle at a waypoint node if it shares some of an existing BIB’s targets.  Target multiplicity and security block interactions. | Not supported.  This block interaction will be implemented in the ION 4.2 release |
| 32 | Check that a BCB can be added to the bundle at a waypoint node if it shares some of an existing BIB’s targets.  Target multiplicity and security block interactions. | Not supported.  This block interaction will be implemented in the ION 4.2 release. |
| 33 | BIB target multiplicity .  Check that a BIB can have multiple targets of different block types. | Not supported.  Target multiplicity will be implemented in the ION 4.2 release. |
| 34 | BCB target multiplicity.  Check that a BCB can have multiple targets of different block types. | Not supported.  Target multiplicity is not supported by the ION Test Security Context or BCB-AES-GCM. |
| 35 | BIB target multiplicity.  Check that a BIB can have multiple targets of the same block type. | Not implemented.  Target multiplicity will be implemented in the ION 4.2 release. |
| 36 | BCB target multiplicity.  Check that a BCB can have multiple targets of the same block type. | Not implemented.  Target multiplicity is not supported by the ION Test Security Context or BCB-AES-GCM. |
| 37 | Multi-bundle test | Passed. |
| 38 | Check that a Security Verifier can identify a missing BIB.  Security operation event associated: sop\_missing\_at\_verifier. | Not supported.  Identification of missing security operations will be implemented in the ION 4.2 release. |
| 39 | Check that a Security Verifier can identify a missing BCB.  Security operation event associated: sop\_missing\_at\_verifier. | Not supported.  Identification of missing security operations will be implemented in the ION 4.2 release. |
| 40 | Exercise the bpsecadmin ‘find’ command. | Passed. |
| 41 | Security context ID key handling. | Passed. |
| 42 | BIB targeting an Extension Block. | Passed. |
| 43 | BCB targeting an Extension Block. | Passed. |
| 44 | Encrypt a large payload (1MB). | Passed. |