# BPSec Python Test Suite

The Johns Hopkins University Applied Physics Laboratory (JHU/APL)

January 2022

# Overview

This Python test suite has been written to test the implementation of BPSec in ION. The included test cases cover the bib-integrity and bcb-confidentiality security services and test features such as target multiplicity, mixed security policy, and uniqueness of security operations.

# 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 the following section.

# Running the Test Suite

To run the python test suite, use the following command:  
 **python3 dotest.py <test>**

Where **<test>** is a number of an individual test to run (tests 1-39 are supported). For example, to run test 1, use the following command:

**python3 dotest.py 1**

Multiple tests can be specified, with each test number separated by a space. For example, to run tests 1, 2, and 5, use the following command:

**python3 dotest.py 1 2 5**

To run one of the supported groups of tests, provide one of the following group names  
 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.

# 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 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. Create a misconfigured security policy rule for the Security Acceptor at ipn:3.1 using the wrong security context (ex: HMAC256). | 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 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. Create a misconfigured security policy rule for the Security Acceptor at ipn:3.1 using the wrong security context (ex: AES512). | 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 | Removed. | N/A | N/A | N/A |
| 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 Primary 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 |

# Test Case Current Results: IOS 4.1.1

|  |  |  |
| --- | --- | --- |
| Test ID | Purpose | Test Status |
| 1 | Test Payload block integrity. | Failing. |
| 2 | Test Payload block confidentiality. | Failing. |
| 3 | Test Primary block integrity. | Failing. |
| 4 | Check that a BCB is not permitted to target a Primary block. | Failing. |
| 5 | BIB behavior at a Security Source, Security Verifier, and Security Acceptor. | Failing. |
| 6 | BCB behavior at a Security Source, Security Verifier, and Security Acceptor. | Failing. |
| 7 | Check that duplicate security operations – in this case, a BIB on the Payload block – are prevented.  Ensure that rule prioritization works. | Failing. |
| 8 | Check that duplicate security operations – in this case, a BCB on the Payload block – are prevented.  Ensure that rule prioritization works. | Failing. |
| 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. | Not implemented.  To be completed for the ION 4.2 release which includes an implementation of the BPSec Default Security Contexts and their variants. |
| 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. | Not implemented.  To be completed for the ION 4.2 release which includes an implementation of the BPSec Default Security Contexts and their variants. |
| 11 | Check that a Security Acceptor can identify a missing BIB.  Security operation event associated: sop\_missing\_at\_acceptor | Failing. |
| 12 | Check that a Security Acceptor can identify a missing BCB.  Security operation event associated: sop\_missing\_at\_acceptor | Failing. |
| 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. |
| 15 | Check that a Security Verifier can identify a corrupted BIB target.  Security operation event associated: sop\_corrupted\_at\_verifier. | Not implemented. |
| 16 | Check that a Security Verifier can identify a corrupted BCB target.  Security operation event associated: sop\_corrupted\_at\_verifier. | Not implemented. |
| 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. | Implemented.  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. | Implemented.  To be used for INB testing.  Enhancement request for additional reporting and granularity of event tracking for this scenario added. |
| 19 | Check that a Security Verifier can identify a misconfigured BIB.  Security operation event associated: sop\_misconfigured\_at\_verifier. | Implemented.  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. | Implemented.  To be used for INB testing. |
| 21 | Removed. | N/A |
| 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. | Failing. |
| 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. | Failing. |
| 24 | Check that the BPSec implementation does not permit a BCB to target another BCB. | Failing. |
| 25 | Check that a BIB and BCB are permitted to share a security target. | Failing. |
| 26 | Check that a BIB is not added at a waypoint node if a BCB already exists with the same target block. | Failing. |
| 27 | Check that the BPSec implementation does not permit a BIB to target another BIB. | Failing. |
| 28 | Check that a BCB is not permitted to target a BIB it does not share a security target with. | Failing. |
| 29 | Check that a BCB can encrypt a BIB if the two security blocks share a security target. | Failing. |
| 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 implemented.  To be completed for 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. | Failing. |
| 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 implemented.  To be completed for the ION 4.2 release. |
| 33 | BIB target multiplicity  Check that a BIB can have multiple targets of different block types. | Failing. |
| 34 | BCB target multiplicity  Check that a BCB can have multiple targets of different block types. | Not implemented. |
| 35 | BIB target multiplicity  Check that a BIB can have multiple targets of the same block type. | Not implemented. |
| 36 | BCB target multiplicity  Check that a BCB can have multiple targets of the same block type. | Not implemented. |
| 37 | Multi bundle test | Failing. |
| 38 | Check that a Security Verifier can identify a missing BIB.  Security operation event associated: sop\_missing\_at\_verifier. | Failing.  Issue has been created. |
| 39 | Check that a Security Verifier can identify a missing BCB.  Security operation event associated: sop\_missing\_at\_verifier. | Failing.  Issue has been created. |