<https://tools.ietf.org/html/draft-ietf-dtn-bpsec-20>

**[0. Terms]**

|  |  |
| --- | --- |
| Bundle Source | Bundle forwarding이 시작되는 지점 |
| Bundle Destination | Bundle forwarding이 종료되고 최종 수신 노드가 있는 지점 |
| Cipher Suite | 무결성, 기밀성을 제공하는 1개 이상의 알고리즘의 집합 |
| Forwarder | DTN에서 bundle을 전송하는 노드 |
| Intermediate Receiver  Waypoint  Next Hop | Forwarder에게서 bundle을 수신하지만, Bundle Destination이 아닌 노드 (또는 그러한 노드의 Node ID) |
| Path | Source에서 Destination으로 가는 Bundle이 지나가기로 계획된 node의 순서 |
| Security Block | Bundle에 있는 BPSec extension block |
| Security Acceptor | 1개 이상의 Security Block을 프로세싱 및 배치하는 bundle node (또는 그러한 노드의 Node ID) |
| Security Context | Security Service 구현을 위한 가정, 알고리즘, 설정, 정책의 집합 |
| Security Operation | Security Target, 즉 OP에 대한 Security Service의 연산   * **각 bundle의 모든 Security Operation은 유일**해야 한다. 즉 하나의 Security Service는 어떤 bundle에서 같은 Security Target에 1번만 적용될 수 있다. |
| Security Service | Security Target에 대한 보호를 제공하는 프로세스 |
| Security Source | Bundle에 Security Block을 추가하는 bundle node |
| Security Target | Security Operation의 일부로 Security Service를 수신하는, bundle 안에 있는 블록 |

**[1. Design Decisions – Block-level Granularity]**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Security Service는 bundle 각각에 적용되는 Security Service를 위해 대해 서로 다른 블록을 허용해야 한다.**   * Bundle에 있는 **각 block은 서로 다른 종류의 정보**를 나타낸다.  |  |  | | --- | --- | | Primary Block | Identification, routing 정보 | | Payload Block | Application 정보 | | Extension Block | Payload 또는 bundle의 프로세싱과 관련된 다양한 정보 |  * 단일 레벨과 종류의 보안을 전체 bundle에 적용하는 것으로는 **각 bundle에 있는 블록이 서로 다른 보안을 필요로 하는 서로 다른 종류의 정보를 나타낼 수 없다**. |

**[2. Design Decisions – Multiple Security Sources]**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BPSec을 구현할 때 bundle 안에 있는 모든 블록에 같은 보안 연산이 적용되어 있을 것이라고 가정하면 절대 안 된다.**   * Bundle protocol은 bundle이 DTN에 존재할 때 **그 bundle의 extension block이 추가되는 것을 허용**한다. * Waypoint가 bundle에 추가할 때…  |  |  | | --- | --- | | 새로운 Extension block | 그 Extension block에는 **Security Service**가 적용될 수 있다.   * 마찬가지로 waypoint는 **존재하는 extension block에 Security Service를 추가**할 수 있다. | | 새로운 Security Service | **그 Waypoint는 그 서비스의 Security Source**이다. | |

**[3. Design Decisions – Mixed Security Policy]**

|  |
| --- |
| **DTN의 node에 대해 강제되는 Security Policy는 서로 다를 수 있다.**   * 어떤 waypoint는 **Security-aware하지 않고, security block을 프로세싱할 수 없을** 수 있다. * 어떤 waypoint는 Bundle destination과 같은 **최종 목적지가 아닌 곳에서도 Security Service에 대한 evaluation이 필요**한 Security Policy를 가질 수 있다. * 어떤 waypoint는 Security block에 대해서 이해하지만 **그것을 프로세싱하는 것을 거부**할 수 있다. |

**[4. Design Decisions – User-defined Security Contexts]**

|  |
| --- |
| **Security Context:** **보안 알고리즘(=Cipher Suites), 그 알고리즘의 사용 정책, 설정값의 조합**   * 서로 다른 context는 Security Service 구현에 대한 서로 다른 알고리즘, 정책, 설정값을 나타낸다. * 사용자는 등록된 Security Context 중 선택할 수 있고, **Security Context parameter를 통해 그 context들을 커스터마이징**할 수 있다. |

**[5. Design Decisions - Deterministic Processing]**

|  |
| --- |
| 모든 노드는 **모든 security block에 대해 동일하고 deterministic한 프로세스 순서**를 따라야 한다.   * 이것은 유연성을 잃는 한이 있더라도 **application과 Security Service의 evaluation에 대한 결정성을 제공**한다. |

**[6. Security Blocks – Block Definitions]**

|  |
| --- |
| **Block Integrity Block (BIB):** **평문의 Security Target에 대한 무결성을 보증**하는 데 사용된다.   * BIB에 있는 무결성 정보는 **BIB Security Source ~ Bundle destination 구간의 bundle path에 있는 어떤 노드에 의해서라도 번역**될 수 있다. * Security-aware waypoint는 **Security Policy와 관련하여 bundle로부터 BIB를 생성하거나 삭제**할 수 있다. * BIB는 BCB에 의해 제공되는 암호문의 무결성 보호에 **사용되지 않는다**.   **Block Confidentiality Block (BCB):** **BCB Security Source가 전송되는 동안 그 내용을 보호**하기 위하여, **Security Target이 BCB Security Source에서 암호화**되었는지를 나타낸다.   * BCB는 네트워크의 bundle destination까지 **Security-aware한 노드에 의해 복호화**된다. |

**[7. Security Blocks – Uniqueness]**

|  |
| --- |
| **Bundle 안에 있는 Security Operation은 반드시 유일해야 한다.**   * 서로 같은 Security Service는 한 bundle에 있는 어떤 security target에 **1회를 초과하여 적용될 수 없다**.   + Security Operation은 하나의 Security Block으로 표현되므로, 이것은 **Security block이 어떻게 bundle에 추가될지의 방법을 제한**한다. * Security Operation은 bundle에서 제거될 수 있다.   + Security Operation이 bundle에서 한번 제거되면 **같은 Security Operation은 새로운 Security Block을 bundle에 추가하는 방법으로 재적용**될 수 있다. * **BCB가 공급하는 어떤 암호문 무결성 메커니즘도 기밀성 서비스의 일부분으로 간주된다. 따라서 BIB가 제공하는 평문 무결성 서비스에서는 유일하다.** * **Notation OP(서비스, 타겟)**를 사용한, **유일성 요구 조건**을 나타내는 예제로는 Signing the payload twice, Signing different blocks, Different Service on same block 등이 있다. |

**[8. Security Blocks – Target Multiplicity]**

|  |
| --- |
| **단일 Security Block은 Bundle에 있는 전체 Security Block의 개수를 줄이는 방법으로써 여러 개의 security operation을 나타낼 수 있다.**   * Security Operation의 집합은 **다음 모든 조건이 참일 때** 단일 Security Block에 의해 나타내어질 수 있다.   + Security Operation에 같은 Security Service가 적용된다.   + Security Operation의 Security Context Parameter들은 모두 동일하다.   + Security Operation에 있는 Security Source들은 모두 동일하다.   + 어떤 Security Operation도 서로 같은 Security Target을 갖지 않는다.   + 어떤 Security Operation도 bundle에 이미 있는 Security Operation과 충돌되지 않는다. |

**[9. Security Blocks – Target Identification]**

|  |
| --- |
| **Security Target은 Security Service에 적용되는, bundle 안에 있는 block이다.**   * 이 Target은 **Security block을 프로세싱할 때 유일하고 명확하게 인식 가능**해야 한다. * Security Block에 있는 Security Target은 **Target Block의 Block Number**로 표현될 수 있어야 한다. |

**[10. Security Blocks – Block Representation]**

|  |
| --- |
| **각 Security Block은 Canonical한 Bundle Block Format을 갖는다.**   * 각 Security Block은 다음과 같은 요소들을 갖는다.   Block Type Code, Block Number, Block Processing Control Flags, CRC type, Block-Type-Specific-Data, CRC field (if present)   * Security Block에 대한 **Security-specific한 정보는 Block-type-specific-data 필드에서 capture**된다. |

**[11. Security Blocks – Abstract Security Block]**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Security Block의 **Security-specific한 부분의 구조는 BIB, BCB 블록에서 동일하다.**  **Abstract Security Block (ASB):** **BIB, BCB Security Block에서의 일반적인 양상을 설명하기 위한 메커니즘**  <ASB의 필드>   |  |  | | --- | --- | | Security Target(s) | **Security Operation에 의해 target되는 블록**   * 각 target block은 유일한 Block Number로 나타내어진다. | | Security Context id | 그 블록에 의해 표현되고 각 security target에 적용되는 **Security Service를 구현하기 위한 Security Context** | | Security Context Flags | Security Block에 **현재 어떤 Optional field가 있는지**를 나타낸다.  (있으면 1, 없으면 0 / 비트 0은 Security Context Parameters Present Flag, 비트 1은 Security Source Present Flag, 비트 2~는 예약됨) | | Security Source (opt.) | **Bundle에 Security Block을 추가한 Endpoint**를 identify한다. | | Security Context  Parameters (opt.) | Security Block에 의해 설명되는 Security Service를 프로세싱할 때 **Security-aware 노드에 제공되어야 하는, 1개 이상의 Security Context Parameter**를 capture한다.   * 단일 parameter는 **Parameter ID, Parameter Value**의 2개의 원소가 있는 CBOR array로 표현되어야 한다. | |
| **NEXT PAGE** |

**[11. Security Blocks – Abstract Security Block]**

|  |
| --- |
| **PREVIOUS PAGE** |
| |  |  | | --- | --- | | Security Results | **Security Service를 Security Block의 Security Target에 적용했을 때의 결과**를 Capture한다.   * Target result로 표현된 CBOR array로 표현되어야 한다. * 개별적인 result는 **result id와 result value의 2개의 튜플이 있는 array로 표현**된다. | |

**[12. Security Blocks – Block Integrity Block (BIB)]**

|  |
| --- |
| **Block Integrity Block (BIB):** 다음과 같은 특징을 갖는 bundle extension block   * Block Type Code는 다음과 같다. (<https://tools.ietf.org/html/rfc6255>)      * Block-type-specific-data 필드는 ASB의 구조를 따른다. * Security Target Field에 나열된 **Security target은 BIB, BCB 등의 Security block을 참조해서는 안 된다.** * Security Context는 **인증 메커니즘 또는 오류 탐지 메커니즘**을 반드시 이용해야 한다. * Security Source의 **EID는 반드시 존재**해야 한다. |

**[13. Security Blocks – Block Confidentiality Block (BCB)]**

|  |
| --- |
| **Block Confidentiality Block (BCB):** 다음과 같은 특징을 갖는 bundle extension block   * Block Type Code는 BIB에서와 같다. * Block Processing Control Flag Value는 **몇 가지 예외를 제외하고 local 정책에서 요구하는 어떤 값이라도** 될 수 있다. * Block-type-specific-data 필드는 **ASB의 구조를 따른다**. * Security Target Field에 나열된 **Security target은 BIB, BCB 등의 Security block을 참조할 수 있다.** * Security Context는 **인증된 암호화를 제공하는 기밀성 암호(AEAD)**를 사용해야 한다. * Cipher Suite에 의해 생성된 추가적인 정보는 Security Result field 또는 Generated Cipher Text field에 위치할 수 있다. * Security Source의 **EID는** **존재할 수 있다.** |

**[14. Security Blocks – Block Interactions]**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Security Block은 서로 독립적으로 설계되었지만, Security target을 서로 공유하는 경우가 있다.**   * **BCB의 Security target과 BIB의 그것이 같은 경우**, Security aware waypoint가 BIB를 활성화할 수 없으면 의도하지 않은 상황이 발생할 수 있다. 예를 들면,   + Security target의 내용이 BCB에 의해 암호화된 경우 등     - 이때는 다음과 같은 규칙을 따라야 한다.       * BCB를 bundle에 추가할 때,  |  |  |  | | --- | --- | --- | | BCB의 Security target 중 | BIB의 Security target과 매칭 | 결과 | | 일부 또는 전체 | 전체 | **BIB도 반드시 암호화** | | 일부 또는 전체 | 일부 | BIB의 **BCB Security Target과 연관된 어떤 Security Result도 BIB에서 제거되고 새로운 BIB에 위치**해야 한다. |  * + - * BIB는 이미 BCB의 Security target인 Security target에 추가될 수 없다. |

**[15. Security Blocks – Parameter and Result Identification]**

|  |
| --- |
| **각 Security Context는 고유한 context parameter와 result를 정의해야 한다.**   * 각각의 정의된 parameter와 result는 **identifier와 value로 구성된 튜플**로 표현된다. * **Identifier는 주어진 Security Context에 대해 유일**해야 하지만 모든 Security Context에 대해 유일하지는 않아도 된다. |

|  |
| --- |
| **[2020.03.04 발표]** |

**[16. Canonical Forms]**

|  |
| --- |
| **Security service는 정보가 Security Source와 Receiving Node에서 어떻게 Cipher Suite로 나타나는지에 대한 일관성과 결정성을 필요로 한다.**   * 예를 들어, Integrity Service에서는 Signature를 생성하거나 인증할 때 **같은 target information은 Cipher suite에 의해 제공**되어야 한다. * Canonical Form은 Security-aware Node에 대한 **Security processing을 위한 Security Context에 대한 입력**을 생성하는 데 사용된다. * BPSec은 bundle block 안에 있는 data field에서 작동한다.   + 이 Canonical form에서 이 field들은 **고유의 CBOR 인코딩**을 포함해야 한다. * 모든 **Non-primary Block은 같은 블록 구조**를 가지고, 몇 가지 예외를 제외하고 <https://tools.ietf.org/html/draft-ietf-dtn-bpsec-20#ref-I-D.ietf-dtn-bpbis> 에서와 같이 canonicalize된다. |

**[17. Security Processing – Blocks Received from Other Nodes]**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Security Block은 Security-aware node에 의해 수신될 때 특정한 순서에 따라 프로세싱되어야 한다.**   * **BIB와 BCB가 security target을 공유**할 때, **BCB가 evaluate된 다음에 BIB가 evaluate**되어야 한다.   [For Receiving BCBs]   * **[A] 수신된 Bundle이** **BCB를 포함**하면, 수신하는 node는 이것이 **BCB의 어떤 Security Operation의 Security acceptor인지 결정**해야 한다.   + 그러면 그 노드는 **데이터를 노드의 application에 전송하거나 bundle을 forwarding**하기 전에 **그 operation을 프로세싱하고 BCB에서 operation-specific한 정보를 제거**해야 한다.     - 보안 연산이 실패하면 target은 보안 정책에 따라 프로세싱되어야 한다. * **[B] 수신된 노드**가…  |  |  | | --- | --- | | Bundle의 destination임 | 노드는 **bundle에 남아 있는 BCB를 복호화**해야 한다. | | Bundle의 destination이 아님 | 노드는 **보안 정책에 따라 해야 한다면 BCB를 프로세싱**해야 한다. |  * **[C] 암호화된 X가 복호화 불가능**하면…  |  |  | | --- | --- | | Payload Block | **Bundle**은 기각되고 더 이상 프로세싱되지 않는다. | | Payload Block이 아닌 것 | **연관된 security target과 그 target에 연관된 모든 security block**은 기각되고 더 이상 프로세싱되지 않는다. |  * **[D] BCB가 복호화**되면 각 Security Target에 대한 **복구된 Plain text는 각 Security Target의 block-type-specific-data 필드의 암호문을 대체**한다.   [For Receiving BIBs]   * **[A]** For Receiving BCBs의 **[A]**에서 **BCB->BIB**로 바꾼 것과 같다. * **[B]** BIB는 어떤 bundle에서 **BIB와 BCB의 Security target이 서로 같을 때는 프로세싱될 수 없다.**   + BIB와 BCB가 Security target을 공유한다는 것은 **그 target은 integrity sign이 된 다음에 암호화되었다**는 것을 의미하므로, **Security target이 BCB에 의해 복호화되기 전까지는 BIB가 verify될 수 없다**. * **[C]** Security-aware node의 보안 정책에서, **노드가 특정한 Security target에 대해 integrity를 적용했고 그러한 BIB이 bundle에 없다**면, 그 노드는 **그 Security Target을 프로세싱**해야 한다. * **[D]** 수신하는 노드가 BIB에 있는 Security Operation의 Security Acceptor가 아니면, 어쨌든 **Corrupt data가 forwarding되는 것을 방지하기 위해 Security Operation을 verify하려고 시도**해야 한다. |

**[18. Security Processing – Bundle Fragmentation and Reassembly]**

|  |
| --- |
| **Node가 bundle payload를 조각낼 필요가 있고 그 bundle에 Security Service가 적용되어 있다면,** [**https://tools.ietf.org/html/draft-ietf-dtn-bpsec-20#ref-I-D.ietf-dtn-bpbis**](https://tools.ietf.org/html/draft-ietf-dtn-bpsec-20#ref-I-D.ietf-dtn-bpbis) **에 제시되어 있는 Fragmentation rule을 따라야 한다.**   * **오직 Payload Block만 fragment**될 수 있고, Security Block이나 다른 Extension Block은 fragment될 수 없다. * Payload block fragmentation의 복잡성 때문에 **무결성과 기밀성 연산은 fragment를 나타내는 bundle에는 적용되지 않는다**. * Payload block fragmentation이 있을 때, Security processing은 **BPSec protocol 외의 다른 메커니즘** 또는 **캡슐화 메커니즘과 같이 작동하는 BPSec block의 적용**에 의해 이루어져야 한다. |

**[19. Key Management]**

|  |
| --- |
| **DTN에서 key information을 생성, 통신 등 관리하는 방법은 매우 많다.**   * 각 DTN을 이용할 때는 key management를 위해 이미 생성된 프로토콜을 따라야 하는 반면, 다른 DTN에 대해서는 새로운 접근 방법을 이용해야 한다. * BPSec은 **key management가 network management와 별도인 것처럼 작동**하고, 이것은 **특정한 key management 전략을 요구하지도, 지정하지도 않는다**고 가정한다. |

**[20. Security Policy Considerations]**

|  |
| --- |
| **BPSec을 구현할 때, 몇 가지의 Policy Decision이 고려되어야 한다.**   * **허용되지 않은 Security Operation의 조합**이 포함된 bundle이 수신되면 BPA는 이것을 어떻게 handle할지 결정해야 한다. * 네트워크 안에 있는 **BPA는 어떤 Security Operation을 bundle에 적용해야 하는지** 이해해야 한다. * Receiver는 **waypoint가 bundle에 보안 연산을 생성하도록 설정**되었고 **수신된 bundle에 이미 Security Operation이 적용**되어 있으면 어떻게 해야 하는지 이해해야 한다. * **BIB를 이미 BCB에 의해 암호화된 Security Target에 추가하는 것은 허용되지 않는다.** 이러한 상황이 발생할 것 같을 때 가능한 정책은 다음과 같다.   + 암호화할 때, **평문의 무결성 signature를 계산하는 Security Context가 선택**되고, **Security Context Result 필드처럼 포함**될 수 있다.   + 암호화된 블록은 **새로운 블록 번호와 주어진 무결성 보호를 포함한 새로운 블록으로 복제**될 수 있다.   + 캡슐화 scheme은, **캡슐화 구조가 더 이상 BCB의 Security Target이 아니고 BIB의 Security Target일 수 있는** Security Target을 캡슐화하는 데 적용될 수 있다. |

**[21. Security Considerations]**

|  |
| --- |
| **DTN에서 bundle은 다양한 환경과 디바이스에서 이동할 수 있고, 따라서 BPSec은 보안 위험이 있다.**   * **Key threat model을 도입**하고 **BPSec 프로토콜의 역할과 책임을 명시**하는 것이 기밀성과 무결성을 보호하기 위해서 필요하다. * **BPSec threat model은 DTN 환경에서 동작하는 BPSec에 대한 위협을 설명**하는 데 집중하므로, 여기서는 **Man-in-the-middle (MITM) 공격 문제**를 다룬다. |

**[22. Security Considerations - Attacker Capabilities and Objectives]**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **BPSec은 bundle이 Source(Alice)에서 Destination(Bob)으로 이동할 때 MITM 공격의 위협으로부터 보호하도록 설계되었다.**   * **MITM node (Mallory)**는 DTN의 Alice와 Bob 사이에 있는 비협조적인 노드이다.   + 능력: DTN에 있는 데이터의 기밀성 또는 무결성을 침해하기 위한, **bundle의 송수신, 수정, 검사 및 생성** * MITM node의 유형은 다음과 같다.  |  |  | | --- | --- | | Unprivileged Node | Mallory는 DTN 환경에서 충분한 정보를 제공받지 못한다. **즉, 공유된 암호 정보에만 접근**할 수 있다. | | Legitimate Node | Mallory는 DTN 환경에서 **Mallory에게 충분한 정보가 제공된 암호 정보(K\_M)에 접근**할 수 있다.   * Mallory가 Alice 또는 Bob일 때(즉 **K\_M == K\_A or K\_M == K\_B**) Mallory는 **Alice 또는 Bob인 것처럼 위장**할 수 있다. 이것은 **그 노드로 가는 트래픽이 각각 암/복호화될 수 있다**는 것을 의미한다. | | Privileged Node | Mallory는 DTN 환경에서 권한이 부여된 노드이며, **Mallory, Alice, Bob에게 충분한 정보가 제공된 암호 정보(K\_M, K\_A, K\_B)에 접근**한다.   * **BPSec은 Mallory를 탐지하거나 제거하는 메커니즘을 제공하지 않으므로** BPSec 개발자가 필요한 기능을 추가적으로 제공해야 한다. * 구현된 BPSec이 **모든 노드**에 대해 **하나의 공유된 암호 정보를 사용한다면, Legitimate Node = Privileged Node**이다. | |

**[23. Security Considerations – Eavesdropping Attacks]**

|  |
| --- |
| **Mallory가 bundle을 수신하면 그 bundle에 포함된 블록의 내용을 검사하고, 보호된 데이터 또는 암호 키 정보를 복구하려고 시도할 수 있다.**   * 이에 대한 BPSec의 보호 메커니즘은, **그것의 내용을 암호화하는 BCB이다. (기밀성)**   + Mallory가 Offline recovery를 할 수 있으므로 **암호화 메커니즘의 암호화가 충분히 복잡해야** 한다. * Eavesdropping attack의 위험성을 평가하기 위해서 **bundle의 lifetime**을 고려해야 하는데, 이것이 **길기 때문에 capture당하고 암호화 알고리즘이 풀릴 가능성이 높다**.   + Mallory는 bundle lifetime뿐만 아니라 **bundle을 저장하여 영구 보관**할 수 있다. |

**[24. Security Considerations – Modification Attacks]**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DTN에서 Alice와 Bob의 중간에 있는 Mallory는 Non-BPSec data인 Primary, Payload Block 등을 포함하여 수신한 bundle을 수정할 수 있다.**   * Mallory는 블록 안에 있는 **데이터의 수정, 교체, 추가, 삭제** 등을 할 수 있다. * BPSec Security Operation은 블록을 bundle에 추가하여 구현되었으므로 **Mallory가 bundle에서 블록을 제거한 것을 감지하거나 수정하는 메커니즘이 없다**. * BPSec에서는 **BIB와 BCB**가 Mallory의 데이터 수정 시도로부터 **데이터의 무결성을 보호하는 메커니즘**을 가지고 있다.  |  |  | | --- | --- | | BIB | 자신의 **Security Target인 다른 블록에 대한 보호**를 제공   * 암호화 메커니즘이 **Collision Attack**에 대응할 수 있도록 충분히 강해야 한다. * Mallory가 **원래 노드에서 사용하는 암호 정보에 접근하여 BIB(K\_A 등)를 생성할 수 없어야** 한다.   **Destination node에서의 BIB validation**의 구현에서도 BIB의 한계점이 존재할 수 있다.   * Mallory가 DTN에 있는 legitimate node이면 K\_A로 생성된 BIB은 **K\_M으로 생성된 BIB으로 교체**되어 Bob에게 전송될 수 있다. | | BCB | BPSec의 구현은 **기대/요구되는 Security Operation**에 대한 설명을 포함한, 네트워크의 **endpoint에 대한 정책** 설정과 결합해야 한다.   * Mallory가 **BCB를 제거했지만 Security Target은 남겨진** 경우, Security Target은 암호화된 상태로 남고, **블록을 복호화하여 destination으로 전송하기 위한 충분한 정보가 없을** 가능성이 생긴다. * Mallory가 **BCB와 그것의 Security Target을 모두 제거**한 경우 bundle에는 **Security Operation을 수행했다는 증거가 남지 않게** 된다. * BCB에 **Verifiable한 무결성 검사**를 제공하기 위하여, BCB에는 **Indistinguishable under adaptive Chosen Ciphertext Attack (IND-CCA2)** 암호화가 필요하다. | |

**[25. Security Considerations – Topology Attacks]**

|  |
| --- |
| **Mallory가 DTN의 MITM 공격자이면 네트워크에서 Mallory에게 오는 어떤 bundle에도 영향을 미칠 수 있다.**   * 이때 Mallory에게는 (1) **not forward** the bundle, (2) forward the bundle as **intended**, (3) forward the bundle to **1 or more specific node within network**의 옵션이 있다. * 패킷의 Re-routing을 포함한 공격은 modification attack의 특별한 경우이다. * **BPSec은 Primary Block의 내용을 암호화하지 못하므로**, 이 상황을 방지하기 위해 다른 방법이 쓰여야 한다. |

**[26. Security Considerations – Message Injection]**

|  |
| --- |
| **Mallory는 새로운 bundle을 만들고 그것을 DTN으로 전송할 수 있다.**   * 이 bundle들은 **원래 bundle의 복사본 또는 약간 수정**된 것이거나(replay attack), 또는 Bundle Protocol, BPSec에 의해 생성된 **완전히 새로운 bundle**이다. * **Bundle Protocol, Storage** 등이 공격 대상이 된다.   + Storage를 공격할 때는 Bundle Protocol 또는 응용 계층 프로토콜에서 동작하는 노드들에 대한 계산 등이 목적이다. * BPSec은 **replay attack 또는 forget message attack**을 방지하기 위하여 **Cipher Suite의 기능에 의존**한다.   + 적절한 암호 메커니즘과 함께 사용되는 **BCB는 특정한 환경에서 replay attack으로부터 보호**한다.   + **BIB**는 **데이터의 유일성을 보장**하는 메커니즘이 포함된 **응용** 계층 데이터를 보호한다. |

**[27. Security Context Considerations – Mandating Security Contexts]**

|  |
| --- |
| **네트워킹 시나리오와 BPSec을 이용하는 노드의 기능이 다양하므로, 모든 가능한 BPSec 구현에 적용되는 하나의 Security Context는 존재하지 않는다.**   * Limited connectivity를 가진, Resource-constrained node에 적합한 Security Context는 well-resourced, well-connected node에는 적합하지 않다. * **BPSec의 구현은 네트워크에 적용되는 Security Context를 지원**해야 한다. * BPSec은 **다양한** Security Context 및 **이후에 정의될** Security Context를 지원한다. |

**[28. Deep Learning/AI IDEA of DTN]**

|  |
| --- |
| <https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20190000830.pdf>  Delay Tolerant Network Routing as a Machine Learning Classification Problem  Rachel Dudukovich, NASA Glenn Research Center Cleveland, Ohio 44135 Email: [rachel.m.dudukovich@nasa.gov](mailto:rachel.m.dudukovich@nasa.gov)  Christos Papachristou, Case Western Reserve University Cleveland, Ohio 44106 Email: [cap2@case.edu](mailto:cap2@case.edu)   * Naïve Bayes, Decision Tree, K-Means 등 다양한 머신러닝 알고리즘을 이용하여 DTN의 라우팅 문제를 해결한다. |