Security Efforts and Extension Block Processing

Angela Hennessy

Laboratory for Telecommunications Sciences (LTS)

Joint Work With:

- Amy Alford
- Kelley Burgin
- Cherita Corbett

Bundle Security Protocol (BSP)

RFC 6257

- Four types of security blocks:
  - Bundle Authentication Block (BAB)
  - Payload Integrity Block (PIB)
  - Payload Confidentiality Block (PCB)
  - Extension Security Bock (ESB)
- Mandatory Ciphersuites:
  - BAB-HMAC
  - PIB-RSA-SHA256
  - PCB-RSA-AES128-PAYLOAD-PIB-PCB
  - ESB-RSA-AES128-EXT

Implementation of PIB, PCB & ESB

- Uses the OpenSSL crypto library
- Mandatory ciphersuites use the Cryptographic Message Syntax (CMS), defined in RFC 5652
- Requires OpenSSL version 1.0.0 or later

Elliptic Curve Ciphersuites

- Internet-Draft uses standard algorithms:

  Digital Signatures: ECDSA

  Key Agreement: ECDH

  Encryption: AES

- Two choices for parameters:

  NIST P-256 (secp256r1)

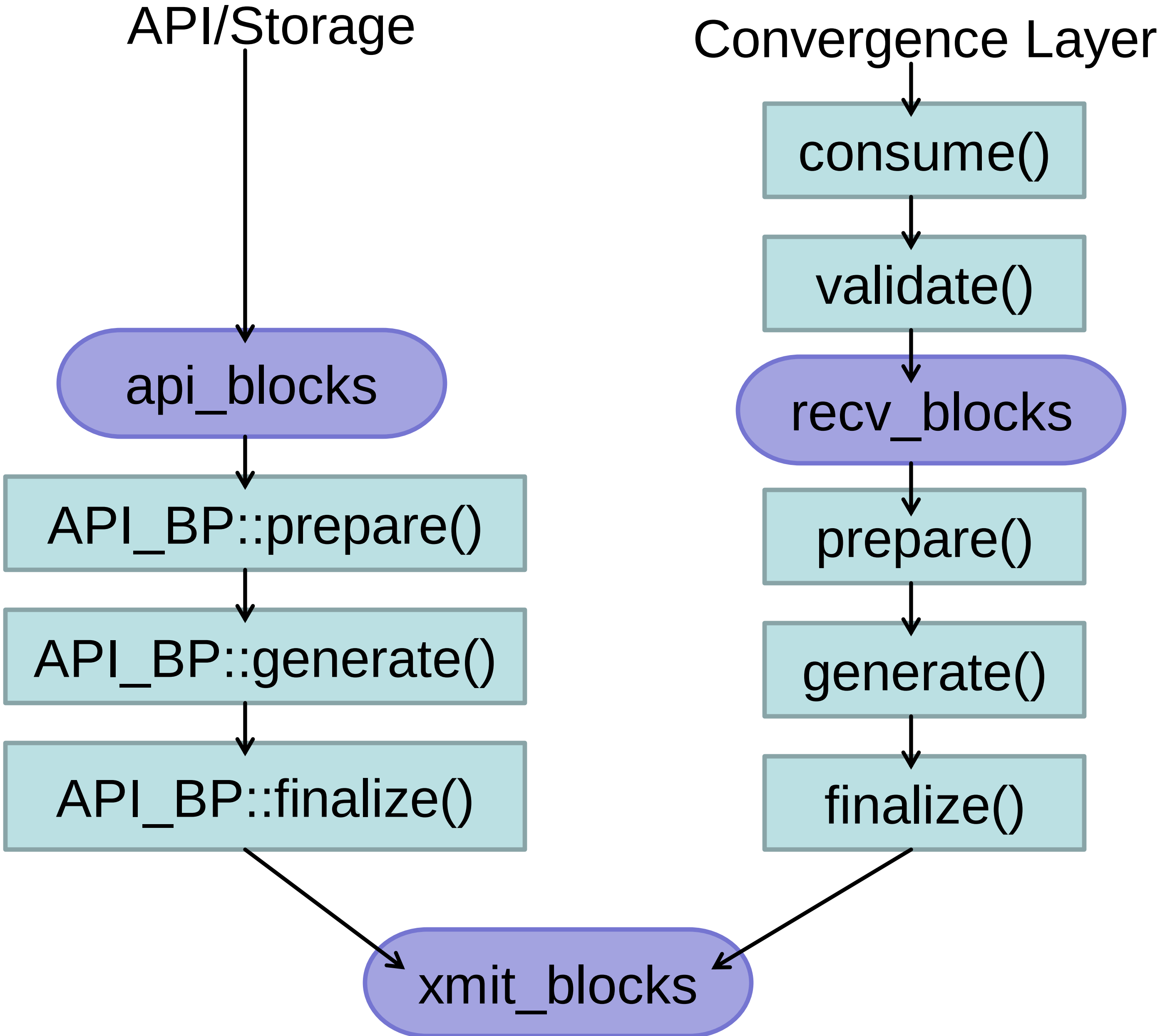
  NIST P-384 (secp384r1)

Extension Integrity Block (EIB)

- There is no method in BSP to digitally sign an extension block
- May want to prevent tampering with information in extension blocks
- Same algorithm as PIB

Extension Block Processing in DTN2

Extension Block Processing in DTN2



Extension Block Processing in DTN2

- Current work-around: change the API bundle_recv_flag from SRC_API to SRC_PEER
- Use API blocks to change configuration of the daemon
- Allows a user to define per-bundle security policies

Extension Block Processing in DTN2
Proposed Changes

- Remove the api_blocks list
- Add blocks to the recv_blocks list, regardless of where the blocks came from
- Move processing of blocks from consume() to validate()
- Simplify existing block processors
  - Age Extension Block
  - Coding Router

Questions?

Security Block Structure

<table>
<tr><td>type</td><td>flags</td><td colspan="2">EID ref list</td></tr>
<tr><td colspan="2">block data len</td><td colspan="2">ciphersuite ID</td></tr>
<tr><td colspan="2">ciphersuite flags</td><td colspan="2">correlator</td></tr>
<tr><td>params len</td><td colspan="3">security params data</td></tr>
<tr><td>result len</td><td colspan="3">security result data</td></tr>
</table>

- Four types of security blocks are defined
- Each block may have several ciphersuites
- Associated to each block is a Security-Source and Security-Destination

Bundle Authentication Block (BAB)

Hop-by-hop Authentication

- Authenticates the bundle along one hop of the communications path
- Covers the entire bundle
- Uses a symmetric key-based algorithm
- Each node shares a secret key with each of its neighbors

Payload Integrity Block (PIB)

End-to-end Authentication

- Authenticates the bundle along the entire communications path
- Source computes an RSA signature with the CMS SignedData content type
- Intermediate nodes can verify the signature

Payload Confidentiality Block (PCB)

End-to-end Encryption

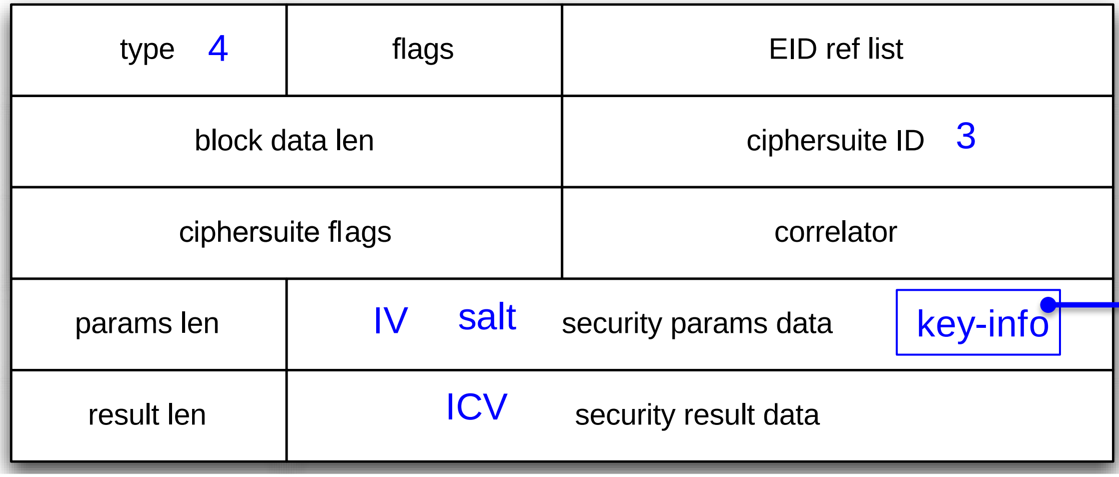
- Encrypts the payload data along the entire communications path
- AES in Galois/Counter Mode for content encryption
- RSA encryption of the AES key with the CMS EnvelopedData content type

Extension Security Block (ESB)

End-to-end Encryption

- Encrypts metadata or extension blocks along the entire communications path
- AES in Galois/Counter Mode for content encryption
- RSA encryption of the AES key with the CMS EnvelopedData content type

Example: PCB & CMS



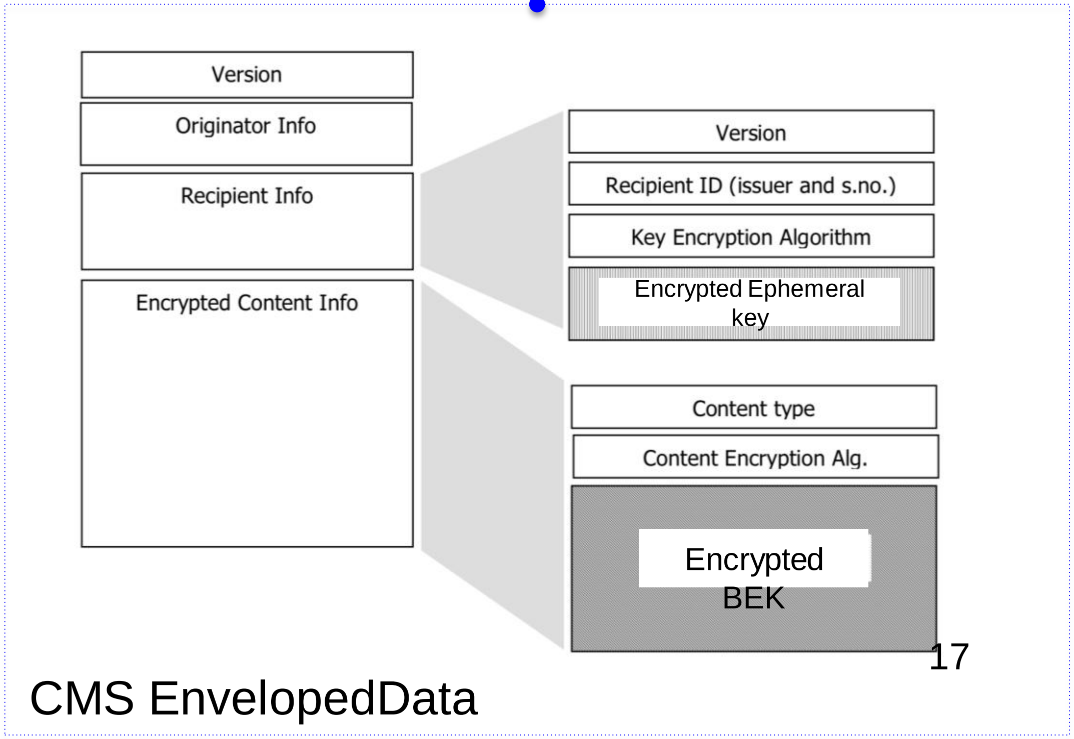
type 4 | flags | EID ref list

block data len | ciphersuite ID 3

ciphersuite flags | correlator

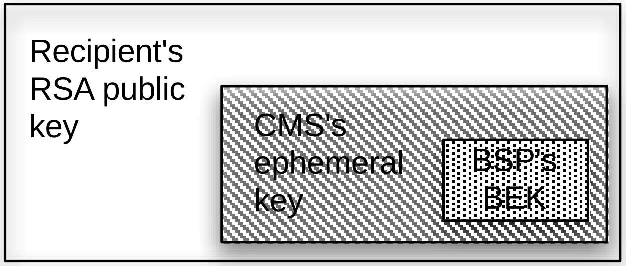
params len | IV salt security params data key-info

result len | ICV security result data



CMS EnvelopedData

Chain of encrypted keys



Key Management Issues

- BSP does not cover key management
- Distributing keys is a challenge in DTNs
- Keys could be pre-placed on each node, or swapped between nodes
- Access to revocation checking services cannot be assumed