Software-Defined Networking (SDN)-based IPsec Flow Protection

(draft-abad-i2nsf-sdn-ipsec-flow-protection-03)

Presenter: Gabriel López-Millán

Rafael Marín-López

(University of Murcia)

SDN-based IPsec: Objectives

- To describe the **architecture** for the SDN-based IPsec management to allow the establishment and management of IPsec security associations from a central point
- To define (so far) the **NSF facing interfaces** required to manage and monitor the IPsec SAs in the NSF from a Security Controller.
  - YANG models are defined for configuration and state data for IPsec management.

Reminder: Two cases

- Case 1) The NSF implements IKE and the IPsec databases: SPD, SAD, and PAD.
  - The Security Controller is in charge of provisioning the NSF with the required information to IKE, the SPD and the PAD.

- Case 2) The NSF only implements the IPsec databases (no IKE implementation).
  - The Security Controller will provide the required parameters to create valid entries in the SPD and the SAD into the NSF.
  - The NSF will have only support for IPsec while automated key management functionality is moved to the controller.

Update (Changes in 03)

- This drafts focuses on: gateway-to-gateway and host-to-host scenarios.
  - Host-to-gateway (roadwarrior) scenario is TBD.
- Improved Case 1 vs Case 2 discussion following comments received.
- It provides YANG configuration data models
  - Case 1 requires IKEv2, SPD and PAD models
  - Case 2 requires SPD and SAD models
  - A single YANG file to represent both cases
    - part of the models are selectively “activated” depending on YANG features (if-feature)

Update- YANG models

- SPD, SAD, PAD models follow the information gathered from RFC 4301
- IKEv2 model has been obtained from the reading of RFC 7296 and using as reference open source implementations (strongswan, libreswan,…)
- Expert review would be appreciated

Update – Implementation notes

- Proof -of-concept
  - NETCONF: southbound protocol
    - Netopeer implementation
    - YANG model
  - Host-to-host and gw-to-gw
  - Case 1, NSF:
    - Strongswan for the IKE implementation (IKE and PAD)
    - VICI API
  - Case 2, NSF:
    - PF_KEYv2 (RFC 2367) for SAD
    - [I-D.pfkey-spd] for SPD
    - XFRM for SAD and SPD (Linux systems)

Software-Defined Networking (SDN)-based IPsec Flow Protection

(draft-abad-i2nsf-sdn-ipsec-flow-protection-03)

Gabriel López-Millán

Rafael Marín-López

(University of Murcia)

YANG Model Trees

Update - SPD model (tree)

```
+--rw spd
|  +--rw spd-entry* [rule-number]
|     +--rw rule-number        uint64
|     +--rw priority?          uint32
|     +--rw names* [name]
|     |  +--rw name-type?   ipsec-spd-name
|     |  +--rw name         string
|     +--rw condition
|     |  +--rw traffic-selector-list* [ts-number]
|     |     +--rw ts-number                uint32
|     |     +--rw direction?               ipsec-traffic-direction
|     |     +--rw local-addresses* [start end]
|     |     |  +--rw start    inet:ip-address
|     |     |  +--rw end      inet:ip-address
|     |     +--rw remote-addresses* [start end]
|     |     |  +--rw start    inet:ip-address
|     |     |  +--rw end      inet:ip-address
|     |     +--rw next-layer-protocol*   ipsec-next-layer-proto
|     |     +--rw local-ports* [start end]
|     |     |  +--rw start    inet:port-number
|     |     |  +--rw end      inet:port-number
|     |     +--rw remote-ports* [start end]
|     |     |  +--rw start    inet:port-number
|     |     |  +--rw end      inet:port-number
|     |     +--rw selector-priority?     uint32
|     +--rw processing-info
|     |  +--rw action          ipsec-spd-operation
|     |  +--rw ipsec-sa-cfg
|     |     +--rw pfp-flag?             boolean
|     |     +--rw extSeqNum?            boolean
|     |     +--rw seqOverflow?          boolean
|     |     +--rw statefulFragCheck?    boolean
|     |     +--rw security-protocol?    ipsec-protocol
|     |     +--rw mode?                 ipsec-mode
|     |     +--rw ah-algorithms
|     |     |  +--rw ah-algorithm*   integrity-algorithm-t
|     |     +--rw esp-algorithms
|     |     |  +--rw authentication*   integrity-algorithm-t
|     |     |  +--rw encryption*       encryption-algorithm-t
|     |     +--rw tunnel
|     |     |     +--rw local?          inet:ip-address
|     |     |     +--rw remote?         inet:ip-address
|     |     |     +--rw bypass-df?      boolean
|     |     |     +--rw bypass-dscp?    boolean
|     |     |     +--rw dscp-mapping?   yang:hex-string
|     |     |     +--rw ecn?            boolean
|     +--rw spd-lifetime
|        +--rw time-soft?       uint32
|        +--rw time-hard?       uint32
|        +--rw time-use-soft?   uint32
|        +--rw time-use-hard?   uint32
|        +--rw byte-soft?       uint32
|        +--rw byte-hard?       uint32
|        +--rw packet-soft?     uint32
|        +--rw packet-hard?     uint32
```

Update - SAD model (tree)

```
+--rw sad {case2}?
|  +--rw sad-entry* [spi]
|     +--rw spi                          ipsec-spi
|     +--rw seq-number?                  uint64
|     +--rw seq-number-overflow-flag?    boolean
|     +--rw anti-replay-window?          uint16
|     +--rw rule-number?                 uint32
|     +--rw local-addresses* [start end]
|     |  +--rw start    inet:ip-address
|     |  +--rw end      inet:ip-address
|     +--rw remote-addresses* [start end]
|     |  +--rw start    inet:ip-address
|     |  +--rw end      inet:ip-address
|     +--rw next-layer-protocol*         ipsec-next-layer-proto
|     +--rw local-ports* [start end]
|     |  +--rw start    inet:port-number
|     |  +--rw end      inet:port-number
|     +--rw remote-ports* [start end]
|     |  +--rw start    inet:port-number
|     |  +--rw end      inet:port-number
|     +--rw security-protocol?           ipsec-protocol
|     +--rw ah-sa
|     |  +--rw integrity-algorithm?   integrity-algorithm-t
|     |  +--rw key?                   string
|     +--rw esp-sa
|     |  +--rw encryption
|     |  |  +--rw encryption-algorithm?   encryption-algorithm-t
|     |  |  +--rw key?                    string
|     |  |  +--rw iv?                     string
|     |  +--rw integrity
|     |  |  +--rw integrity-algorithm?   integrity-algorithm-t
|     |  |  +--rw key?                   string
|     |  +--rw combined
|     |     +--rw combined-algorithm?   combined-algorithm-t
|     +--rw sa-lifetime
|     |  +--rw time-soft?       uint32
|     |  +--rw time-hard?       uint32
|     |  +--rw time-use-soft?   uint32
|     |  +--rw time-use-hard?   uint32
|     |  +--rw byte-soft?       uint32
|     |  +--rw byte-hard?       uint32
|     |  +--rw packet-soft?     uint32
|     |  +--rw packet-hard?     uint32
|     |  +--rw action?          lifetime-action
|     +--rw mode?                          ipsec-mode
|     +--rw statefulfragCheck?             boolean
|     +--rw dscp?                          yang:hex-string
|     +--rw tunnel
|     |  +--rw local?          inet:ip-address
|     |  +--rw remote?         inet:ip-address
|     |  +--rw bypass-df?      boolean
|     |  +--rw bypass-dscp?    boolean
|     |  +--rw dscp-mapping?   yang:hex-string
|     |  +--rw ecn?            boolean
|     +--rw path-mtu?                      uint16
|     +--rw encap
|        +--rw espinudp?   boolean
|        +--rw sport?      inet:port-number
|        +--rw dport?      inet:port-number
|        +--rw oaddr?      inet:ip-address
rpcs:
+---x sadb_register
   +---w input
   |  +---w base-list* [version]
   |     +---w version       string
   |     +---w msg_type?     sadb-msg-type
   |     +---w msg_satype?   sadb-msg-satype
   |     +---w msg_seq?      uint32
   +--ro output
      +--ro base-list* [version]
      |  +--ro version       string
      |  +--ro msg_type?     sadb-msg-type
      |  +--ro msg_satype?   sadb-msg-satype
      |  +--ro msg_seq?      uint32
      +--ro algorithm-supported*
         +--ro authentication
         |  +--ro name?       integrity-algorithm-t
         |  +--ro ivlen?      uint8
         |  +--ro min-bits?   uint16
         |  +--ro max-bits?   uint16
         +--ro encryption
            +--ro name?       encryption-algorithm-t
            +--ro ivlen?      uint8
            +--ro min-bits?   uint16
            +--ro max-bits?   uint16

notifications:
  +---n spd-expire
  |  +--ro index?   uint64
  +---n sadb_acquire
  |  +--ro state    uint32
  +---n sadb_expire
     +--ro state    uint32
```

Update - PAD model (tree)

```
+--rw pad {case1}?
   +--rw pad-entries* [pad-entry-id]
      +--rw pad-entry-id              uint64
      +--rw (identity)?
      |  +--:(ipv4-address)
      |  |  +--rw ipv4-address?            inet:ipv4-address
      |  +--:(ipv6-address)
      |  |  +--rw ipv6-address?            inet:ipv6-address
      |  +--:(fqdn-string)
      |  |  +--rw fqdn-string?             inet:domain-name
      |  +--:(rfc822-address-string)
      |  |  +--rw rfc822-address-string?   string
      |  +--:(dnX509)
      |  |  +--rw dnX509?                  string
      |  +--:(id_key)
      |     +--rw id_key?                  string
      +--rw pad-auth-protocol?       auth-protocol-type
      +--rw auth-method
      +--rw auth-m?          auth-method-type
      +--rw pre-shared
      |  +--rw secret?   string
      +--rw rsa-signature
      +--rw key-data?    string
      +--rw key-file?    string
      +--rw ca-data*     string
      +--rw ca-file?     string
      +--rw cert-data?   string
      +--rw cert-file?   string
      +--rw crl-data?    string
      +--rw crl-file?    string
```

Update - IKE model (tree)

```
+--rw ikev2 {case1}?
|  +--rw ike-connection
|     +--rw ike-conn-entries* [conn-name]
|        +--rw conn-name          string
|        +--rw autostartup        boolean
|        +--rw nat-traversal?     boolean
|        +--rw version?           enumeration
|        +--rw phase1-lifetime    uint32
|        +--rw phase1-authby      auth-method-type
|        +--rw phase1-authalg*    integrity-algorithm-t
|        +--rw phase1-encalg*     encryption-algorithm-t
|        +--rw dh_group           uint32
|        +--rw local
|        |  +--rw (my-identifier-type)?
|        |  |  +--:(ipv4)
|        |  |  |  +--rw ipv4?            inet:ipv4-address
|        |  |  +--:(ipv6)
|        |  |  |  +--rw ipv6?            inet:ipv6-address
|        |  |  +--:(fqdn)
|        |  |  |  +--rw fqdn?            inet:domain-name
|        |  |  +--:(dn)
|        |  |  |  +--rw dn?              string
|        |  |  +--:(user_fqdn)
|        |  |     +--rw user_fqdn?       string
|        |  +--rw my-identifier    string
|        +--rw remote
|        |  +--rw (my-identifier-type)?
|        |  |  +--:(ipv4)
|        |  |  |  +--rw ipv4?            inet:ipv4-address
|        |  |  +--:(ipv6)
|        |  |  |  +--rw ipv6?            inet:ipv6-address
|        |  |  +--:(fqdn)
|        |  |  |  +--rw fqdn?            inet:domain-name
|        |  |  +--:(dn)
|        |  |  |  +--rw dn?              string
|        |  |  +--:(user_fqdn)
|        |  |     +--rw user_fqdn?       string
|        |  +--rw my-identifier    string
|        +--rw local-addrs        inet:ip-address
|        +--rw remote-addr        inet:ip-address
|        +--rw pfs_group?         uint32
|        +--rw phase2-lifetime    uint32
|        +--rw phase2-authalg*    integrity-algorithm-t
|        +--rw phase2-encalg*     encryption-algorithm-t
```