Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 11, 2018

Y. Dong
L. Xia
Huawei

September 07, 2017

The Data Model of Network Infrastructure Device Control Plane Security Baseline
draft-dong-sacm-nid-cp-security-baseline-00

Abstract

This document is one of the companion documents which describes the control plane security baseline YANG output for network infrastructure devices. The other parts of the whole document series [I-D.ietf- xia-sacm-nid-dp-security-baseline], [I-D.ietf-lin-sacm-nid-mp-security-baseline], [I-D.ietf-xia-sacm-nid-app-infr-layers-security-baseline] cover other parts of the security baseline for network infrastructure device in data plane, management plane, application layer and infrastructure layer respectively.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 11, 2018.

Table of Contents

1. Introduction

1.1. Objective

Nowdays network infrastructure devices such as switches, routers, and firewalls are always under the attack of the well-known network security threats which are sammrized in [I-D.ietf-xia-sacm-dp-security-profile]. Hence it is significant to ensure that the devices in a specific network meet the minimal security requirements according to their intended functions. In this case, the concept of security baseline for the network infrastructure device has been proposed in the above mentioned draft [I-D.ietf-xia-sacm-dp-security-profile] as well. The security baseline refers to the basic and compulsory capabilities of identifying the possible threats and vulnerabilities in the device itself, and enfocing the security hardening measurement. And it could be set to benchmark the security posture of an individual network device.

Basically, the overall security baseline of a particular network infrastructure device can be designed and deployed into three different layers, namely the application layer, the network layer, and the infrastructure layer. Moreover, the network layer security baseline is further classified into data plane, control plane, and management plane. In this document, we focus on the designation of data model for control plane security baseline while the security baseline of other layers and planes are proposed in the companion documents.

The control plane security basedline focus on the control signaling security of the network infrastructure device. The aim is to protect the normal information exchange between devices against various attcks (i.e. eavesdropping, tampering, spoofing and flooding attack) and restrict the malicious control signaling, for ensuring the correct network topology and forwading behavior.

1.2. Security Baseline Data Model Design

The security baseline of a certain device is dependent on many factors including but not limited to the different device types (i.e., router, switch, firewall) and their corresponding security features supported, and the specific security requirements of network operators. Owning to such a number of variations, it is impossible to design a comprehensive set of baseline for all devices. This document and the companion ones are going to propose the most important and universal points of them. More points can be added in future following the data model scheme specified in this document.

[I-D.ietf-birkholz-sacm-yang-content] defines a method of constructing the YANG data model scheme for the security posture assessment of the network infrastructure device by brokering of YANG push telemetry via SACM statements. The basic steps are:

- use YANG push mechanism[I-D.ietf-netconf-yang-push]to collect the created streams of notifications (telemetry) [I-D.ietf-netconf-subscribed-notifications]providing SACM content on SACM data plane, and the filter expressions used in the context of YANG subscriptions constitute SACM content that is imperative guidance consumed by SACM components on SACM management plane;

- then encapsulate the above YANG push output into a SACM Content Element envelope, which is again encapsulated in a SACM statement envelope;

- lastly, publish the SACM statement into a SACM domain via xmpp-grid publisher.

In this document, we follow the same way as [I-D.ietf-birkholz-sacm-yang-content] to define the YANG output for network infrastructure device security baseline posture based on the SACM information model definition [I-D.ietf-sacm-information-model].

1.3. Summary

The following contents propose part of the security baseline YANG output for network infrastructure device: control plane security baseline. The companion documents [I-D.ietf- xia-sacm-nid-dp-security-baseline], [I-D.ietf-lin-sacm-nid-mp-security-baseline], [I-D.ietf-xia-sacm-nid-app-infr-layers-security-baseline] cover other parts of the security baseline YANG output for network infrastructure device respectively: control plane security baseline, management plane security baseline, application layer and infrastructure layer security baseline.

2. Terminology

2.1. Key Words

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2.2. Definition of Terms

This document uses the terms defined in [I-D.draft-ietf-sacm-terminology].

3. Tree Diagrams

A simplified graphical representation of the data model is used in this document. The meaning of the symbols in these diagrams is as follows:

- Brackets "[" and "]" enclose list keys.
- Abbreviations before data node names: "rw" means configuration (read-write) and "ro" state data (read-only).
- Symbols after data node names: "?" means an optional node and "*" denotes a "list" and "leaf-list".
- Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (":").

- Ellipsis ("...") stands for contents of subtrees that are not shown.

4. Data Model Structure

A large amount of control protocols such as the typical TCP/IP stack and BGP in the control plane of network infrastructure device provide many operational services (i.e. farwording behavior control). These control protocols could be either the target under attack or the medium to attack the devices. The security baseline of several widely used protocols are specified in this section.

4.1. BGP

In a BGP network, TCP is always selected as the transport layer protocol. Thus it always subject to most of the attacks that targeting TCP-based protocols. In order to secure the BGP network, three types of functions, namely the GTSM, the RPKI, and the BGP peer connection authentication, could be configured in network device. This section specifies the authentication and RPKI configurations. The GTSM is summarized in another individual section together with some other protocols that all supports GTSM.

Various kinds of authentication techniques are able to be used for securing the TCP connections between BGP neighbors. They only allows the authorized peers to establish neighbor relationship with local device so that the information exchanged between the BGP neighbors via the TCP connection cannot be altered.

The Resource Public Key Infrastructure (RPKI) is usually applied in a network equipt with a RPKI server to secure the inter-domain BGP routing. The device is required to establish a connection to the RPKI server and then downloads or updates the Route Origin Authorizations (ROAs), which links certain IP prefixes or prefix range with an autonomous system (AS), from the RPKI server. After that, the received BGP route information is validated against the downloaded/updated ROAs to verify whether the BGP prefixe originates from the expected AS.

```
 module: bgp-sec-config
     +--rw bgp-rpki
     |  +--rw bgp-rpki-session-config* [session-ipv4-addr]
     |  |  +--rw session-ipv4-addr        ipv4-address
     |  |  +--rw port-number              unit16
     |  |  +--rw cipher-password?         string
     |  |  +--rw aging-time?              unit32
     |  |  +--rw refresh-time?            unit16
     |  |  +--rw rpki-limit?
```

```
|  |     +--rw limit                unit32
|  |     +--rw (action-type)
|  |        +--:(alert)
|  |        |  +--rw enable         boolean
|  |        +--:(idle-forever)
|  |        |  +--rw enable         boolean
|  |        +--:(idle-timeout)
|  |           +--rw timeout        unit16
|  +--rw origin-as-validate-utilization
|     +--rw origin-validation-enable boolean
|     +--rw origin-as-validate       boolean
|     +--rw allow-invalide           boolean
|     +--rw (peer-identification-method)
|        +--:(group)
|        |  +--rw peer-group* [group-name]
|        |     +--rw group-name      string
|        |     +--rw advertise-enable boolean
|        +--:(ip)
|           +--rw peer-ip* [ipv4-addr]
|              +--rw ipv4-addr     ipv4-address
|              +--rw advertise-enable boolean
+--rw bgp-authentication* [bgp-as-number]
   +--rw bgp-as-number             unit16
   +--rw (peer-identification-method)
      +--:(group)
      |  +--rw peer-group* [group-name]
      |     +--rw group-name       string
      |     +--rw (authentication-method)
      |        +--:(md5)
      |        |  +--rw password-type:{plain|cipher} enumeration
      |        |  +--rw password-text  string
      |        +--:(keychain)
      |           +--rw keychain-name
      +--:(ip)
         +--rw peer-ip* [ipv4-addr]
            +--rw ipv4-addr        inet-type:ipv4-address
            +--rw (authentication-method)
               +--:(md5)
               |  +--rw password-type:{plain|cipher} enumeration
               |  +--rw password-text  string
               +--:(keychain)
                  +--rw keychain-name  string
```

4.2. OSPF

There are a number of ways for spoofing procotol packet to attack OSPF protocol. One possible scenario is that the rogue device inject manipulated routing information to cause a Denial-of-Service attack.

Authentication has been demonstrated as a powerful tool to identify and drop these spoofing packets to protect OSPF protocol and secure the connection between the OSPF neighbors. A widely range of authentication methods can be deployed in a network device such as MD5, HAMC-MD5, and keychain. As shown in the following tree diagram, the authentication can be deployed in either area or interface basis.

```
 module:ospf-sec-config
     +--rw ospf-authentication
        +--rw area-authentication* [area-id]
        |  +--rw area-id              unit16
        |  +--rw (authentication-method)
        |     +--:(simple-authen)
        |     |  +--rw password-type:{plain|cipher} enumeration
        |     |  +--rw password-text  string
        |     +--:(md5-hmac-authen)
        |     |  +--rw sub-mode:
        |     |  |      {md5|hmac-md5|hmac-sha256} enumeration
        |     |  +--rw password-type   enumeration
        |     |  +--rw password-text   string
        |     +--:(keychain-authen)
        |        +--rw keychain-name   string
        +--rw interface-authentication* [interface-number]
           +--rw interface-type        enumeration
           +--rw interface-number      unit8
           +--rw (authentication-method)
              +--:(simple-authen)
              |  +--rw password-type   enumeration
              |  +--rw password-text   string
              +--:(md5-hmac-authen)
              |  +--rw sub-mode        enumeration
              |  +--rw password-type   enumeration
              |  +--rw password-text   string
              +--:(keychain-authen)
                 +--rw keychain-name   string
```

4.3. IS-IS

IS-IS optional checksum function adds the a checksum TLV in SNP and hello packet. The device firstly check the correctness of checksum TVL when it receive the packet. It secure the data in data link layer.

IS-IS authentication encapsulate the authentication information in hello packet, LSP packet, and SNP packet. Only the packets passed the verification will be further processed. The IS-IS authentication is mainly used to secure packet in network layer.

```
    module:isis-sec-config
        +--rw isis-optional-checksum
        |  +--rw enable                    boolean
        +--rw isis-authentication
           +--rw area-authentication* [process-id]
           |  +--rw process-id          unit32
           |  +--rw (authentication-method)
           |     +--:(simple)
           |     |  +--rw authen-password-mode:{op|osi} enumeration
           |     |  +--rw password-type:{plain|cipher}  enumeration
           |     |  +--rw password-text                 string
           |     +--:(md5)
           |     |  +--rw authen-password-mode:{op|osi} enumeration
           |     |  +--rw password-type:{plain|cipher}  enumeration
           |     |  +--rw password-text                 string
           |     +--:(keychain)
           |     |  +--rw keychain-name                 string
           |     +--:(hmac-sha256)
           |     |  +--rw key-id                        unit16
           |     |  +--rw password-type:{plain|cipher}  enumeration
           |     |  +--rw password-text                 string
           |     +--rw snp-packet:
           |     |      {authentication-avoid|send-only} enumeration
           |     +--rw all-send-only?                   boolean
           +--rw domain-authentication* [process-id]
           |  +--rw process-id                      unit32
           |  +--rw (authentication-method)
           |     +--:(simple)
           |     |  +--rw authen-password-mode:{op|osi} enumeration
           |     |  +--rw password-type:{plain|cipher}  enumeration
           |     |  +--rw password-text                 string
           |     +--:(md5)
           |     |  +--rw authen-password-mode:{op|osi} enumeration
           |     |  +--rw password-type:{plain|cipher}  enumeration
           |     |  +--rw password-text                 string
           |     +--:(keychain)
           |     |  +--rw keychain-name                 string
           |     +--:(hmac-sha256)
           |     |  +--rw key-id                        unit16
           |     |  +--rw password-type:{plain|cipher}  enumeration
           |     |  +--rw password-text                 string
           |     +--rw snp-packet                       enumeration
           |     +--rw all-send-only?                   boolean
           +--rw interface-authentication* [interface-number]
              +--rw interface-type                  enumeration
              +--rw interface-number                pub-type:ifNum
              +--rw (authentication-method)
                 +--:(simple)
```

```
                |  +--rw authen-password-mode:{op|osi} enumeration
                |  +--rw password-type:{plain|cipher}  enumeration
                |  +--rw password-text                 string
                +--:(md5)
                |  +--rw authen-password-mode:{op|osi} enumeration
                |  +--rw password-type:{plain|cipher}  enumeration
                |  +--rw password-text                 string
                +--:(keychain)
                |  +--rw keychain-name                 string
                +--:(hmac-sha256)
                |  +--rw key-id                        unit16
                |  +--rw password-type:{plain|cipher}  enumeration
                |  +--rw password-text                 string
                +--rw authen-level?:{level1|level2} enumeration
                +--rw send-only?                    boolean
```

4.4. MPLS

RSVP authentication is suggested to configure in the device in order to improve the network security and protect the local device against the malicious attack. It prevent the establishment of illegal RSVP peer connection in the following situation

The peer was unauthorized to establish connection with local device;

The attacker establish connection with locol device via spoofing RSVP packet.

Furthermore, it introduce a few enhancement to verify the lifetime, handshake and message window size for protection of RSVP against the playback attack and the termination of authentication relationships caused by packet out of order problem.

As shown in the tree diagram, the LDP also support MD5 and keychain authentication.

```
module:mpls-sec-config
   +--rw rsvp-sec-config
   |  +--rw rsvp-authentication
   |     +--rw interface-authentication
   |     |  +--rw interface-authen* [interface-number]
   |     |     +--rw interface-type                 enumeration
   |     |     +--rw interface-number               pub-type:ifNum
   |     |     +--rw (authentication-method)
   |     |        +--:(md5)
   |     |           |  +--rw password-type:{plain|cipher} enumeration
   |     |           |  +--rw password-text                string
```

```
|     |        +--:(keychain)
|     |        |  +--rw keychain-name               string
|     |        +--rw life-time?             yang-type:timestamp
|     |        +--rw handshake-enable?              boolean
|     |        +--rw window-size?                   unit8
|     +--rw peer-authentication
|        +--rw peer-authen* [peer-addr]
|           +--rw peer-addr                  inet-type:ip-address
|           +--rw (authentication-method)
|              +--:(md5)
|              |  +--rw password-type:{plain|cipher} enumeration
|              |  +--rw password-text                string
|              +--:(keychain)
|              |  +--rw keychain-name                string
|              +--rw challenge-maximum-miss-times?   unit8
|              +--rw challenge-retrans-interval?     unit16
|              +--rw life-time?             yang-type:timestamp
|              +--rw handshake-enable?              boolean
|              +--rw window-size?                   unit8
+--rw ldp-sec-config
   +--rw ldp-authentication
   +--rw (authentication-method)
      +--:(keychain)
      |  +--rw (authen-object)
      |     +--:(peer-single)
      |     |  +--rw single-peer-authen* [peer-id]
      |     |     +--rw peer-id                 dotted decimal
      |     |     +--rw keychain-name               string
      |     +--:(peer-group)
      |     |  +--rw group-peer-authen* [ip-prefix-name]
      |     |     +--rw ip-prefix-name              string
      |     |     +--rw keychain-name               string
      |     +--:(peer-all)
      |        +--rw keychain-name                 string
      |        +--rw exclude-peer-id?           dotted decimal
      +--:(md5)
         +--rw (authen-object)
            +--:(peer-single)
            |  +--rw single-peer-authen* [peer-lsr-id]
            |     +--rw peer-lsr-id               dotted decimal
            |     +--rw password-type:{plain|cipher} enumeration
            |     +--rw password-text             string
            +--:(peer-group)
            |  +--rw group-peer-authen* [ip-prefix-name]
            |     +--rw ip-prefix-name            string
            |     +--rw password-type:{plain|cipher} enumeration
            |     +--rw password-text             string
            +--:(peer-all)
```

```
                  +--rw password-type:{plain|cipher}  enumeration
                  +--rw password-text                 string
```

4.5. Keychain

Authentication is a widely used technique to ensure the packet information are not been changed/altered by attackers. It requires the information sender and receiver to share the authentication information including the key and algorithm. In addition, the key pairs cannot be delivered in the network (symmetric). However, in order to improve the its reliability, the encryption algorithm and the keys have to be renewed dynamically. It is a complicated and time consuming process to change the keys and algorithm for all the used protocols manually. The keychain provide an solution to renew the authentication keys and algorithm periodically in a dynamic fashion.

```
module:keychain-config
    +--rw keychain-config* [keychain-name]
       +--rw keychain-name               string
       +--rw keychain-mode:
       |     {absolute|periodic|daily|
       |      weekly|monthly|yearly}    enumeration
       +--rw receive-tolerance?
       |  +--:(finite)
       |  |  +--rw tolerance-value              unit16
       |  +--:(infinite)
       |     +--rw infinite-enable              boolean
       +--rw time-mode:{utc|lmt}                enumeration
       +--rw digest-length?                     boolean
       +--rw keychain-id* [key-id]
          +--rw key-id                          unit8
          +--rw keychain-string-type:{plain|cipher} enumeration
          +--rw keychain-string-text            string
          +--rw keychain-algorithm:
          |    {hmac-md5|hmac-sha-256|hmac-sha1_12|
          |     hmac-sha1_20|md5|sha-1|sha-256} enumeration
          +--rw default-key-id?                  unit8
          +--rw (send-time-mode)
          |  +--:(absolute)
          |  |  +--rw start-time           yang-type:timestamp
          |  |  +--rw start-date         yang-type:date-and-time
          |  |     +--rw (count-type)
          |  |        +--:(duration)
          |  |        |  +--rw (finite-or-infinite)
          |  |        |     +--:(finite)
          |  |        |     |  +--rw duration-value  unit32
          |  |        |     +--:(infinite)
```

```
|  |        |        +--rw infinite-enable boolean
|  |        +--:(end)
|  |           +--rw end-time    yang-type:timestamp
|  |           +--rw end-date    yang-type:date-and-time
|  +--:(periodic-daily)
|  |  +--rw start-time           yang-type:timestamp
|  |  +--rw end-time             yang-type:timestamp
|  +--:(periodic-weekly)
|  |  +--rw (count-type)
|  |     +--:(continues)
|  |     |  +--rw start-day-name  enumeration
|  |     |  +--rw end-day-name    enumeration
|  |     +--:(discrete)
|  |        +--rw day-name*       enumeration
|  +--:(periodic-montly)
|  |  +--rw (count-type)
|  |     +--:(continues)
|  |     |  +--rw start-date    yang-type:date-and-time
|  |     |  +--rw end-date      yang-type:date-and-time
|  |     +--:(discrete)
|  |        +--rw date*         yang-type:date-and-time
|  +--:(periodic-yearly)
|     +--rw (count-type)
|        +--:(continues)
|        |  +--rw start-month     enumeration
|        |  +--rw end-month       enumeration
|        +--:(discrete)
|           +--rw month*          enumeration
+--rw (receive-time-mode)
   +--:(absolute)
   |  +--rw start-time            yang-type:timestamp
   |  +--rw start-date         yang-type:date-and-time
   |     +--rw (count-type)
   |        +--:(duration)
   |        |  +--rw (finite-or-infinite)
   |        |     +--:(finite)
   |        |     |  +--rw duration-value  unit32
   |        |     +--:(infinite)
   |        |        +--rw infinite-enable boolean
   |        +--:(end)
   |           +--rw end-time    yang-type:timestamp
   |           +--rw end-date  yang-type:date-and-time
   +--:(periodic-daily)
   |  +--rw start-time         yang-type:timestamp
   |  +--rw end-time           yang-type:timestamp
   +--:(periodic-weekly)
   |  +--rw (count-type)
   |     +--:(continues)
```

```
|     |  +--rw start-day-name   enumeration
|     |  +--rw end-day-name     enumeration
|     +--:(discrete)
|        +--rw day-name*        enumeration
+--:(periodic-montly)
|  +--rw (count-type)
|     +--:(continues)
|     |  +--rw start-date    yang-type:date-and-time
|     |  +--rw end-date      yang-type:date-and-time
|     +--:(discrete)
|        +--rw date*         yang-type:date-and-type
+--:(periodic-yearly)
   +--rw (count-type)
      +--:(continues)
      |  +--rw start-month    enumeration
      |  +--rw end-month      enumeration
      +--:(discrete)
         +--rw month*         enumeration
```

4.6. GTSM

Attackers send a large amount of forging packets to a target network device. Then the forging packets are delivered to the cpu straigtforward when the destinations are correctly checked. The CPU will be overloaded owning to processing such a number of protocol packets. In order to protect the CPU against the CPU utilization attack, a GTSM (generized TTL security mechanism) function is configured to check the TTL (time to live) in the IP head. The packets will send to cpu for further processing only if the TTL number is whithin a pre-defined range.

As shown in the three diagram in the following figure, the GTSM function is configured separately for individual procotols. Each of the protocols, even each list instances in a protocol, has its own pre-defined TTL range.

```
    module:gtsm
        +--rw gtsm-config
           +--rw default-gtsm-action:{drop|pass}
           +--rw bgp-gtsm* [bgp-as-number]
           |  +--rw bgp-as-number             unit32
           |  +--rw (peer-identification-method)
           |     +--:(group)
           |     |  +--rw peer-group* [group-name]
           |     |     +--rw group-name       string
           |     |     +--rw valid-ttl-hops   unit16
           |     +--:(ip)
           |        +--rw peer-ip*  [ipv4-addr]
           |           +--rw ipv4-addr   inet-type:ipv4-address
           |           +--rw valid-ttl-hops   unit8
           +--rw ospf-gtsm* [vpn-instance-name]
           |  +--rw vpn-instance-name       string
           |  +--rw valid-ttl-hops           unit16
           +--rw mpls-ldp-gtsm* [peer-ip-addr]
           |  +--rw peer-ip-addr          inet-type:ip-address
           |  +--rw valid-ttl-hops            unit16
           +--rw rip-gtsm* [vpn-instance-name]
              +--rw vpn-instance-name?      string
              +--rw valid-ttl-hops          unit16
```

5. Network Infrastructure Device Security Baseline Yang Module

TBD

6. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

7. Security Considerations

TBD.

8. Acknowledgements

TBD

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

9.2. Informative References

[I-D.ietf-netconf-subscribed-notifications] Voit, E., Clemm, A., Prieto, A., Nilsen-Nygaard, E., and A. Tripathy, "Custom Subscription to Event Notifications", draft-ietf-netconf-subscribed-notifications-03 (work in progress), July 2017.

[I-D.ietf-netconf-yang-push] Clemm, A., Voit, E., Prieto, A., Tripathy, A., Nilsen-Nygaard, E., Bierman, A., and B. Lengyel, "Subscribing to YANG datastore push updates", draft-ietf-netconf-yang-push-08 (work in progress), August 2017.

[I-D.ietf-sacm-information-model] Waltermire, D., Watson, K., Kahn, C., Lorenzin, L., Cokus, M., Haynes, D., and H. Birkholz, "SACM Information Model", draft-ietf-sacm-information-model-10 (work in progress), April 2017.

Authors' Addresses

Yue Dong
Huawei

Email: dongyue6@huawei.com

Liang Xia
Huawei

Email: frank.xialiang@huawei.com