

Internet Draft
<[draft-chen-rtgwg-key-table-yang-00.txt](#)>
Intended Status: Standards Track
Expires in 6 months

I. Chen
Ericsson

July 1, 2015

YANG Data Model for [RFC 7210](#) Key Table
<[draft-chen-rtgwg-key-table-yang-00.txt](#)>

Status of this Memo

Distribution of this memo is unlimited.

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), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on date.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Internet Draft

Key Table YANG

July 1, 2015

Abstract

[RFC 7210](#) defines a key table that consists of cryptographic keys that stores information for many different types of routing protocols to ensure message security. This document defines a YANG data model that represents the key table defined in [RFC 7210](#), with the information necessary for YANG and NETCONF to provide routing protocol authentication.

Table of Contents

1.	Introduction	3
1.1	Terminology	3
1.2	Tree Diagram	3
2.	Design of the Data Model	4
3.	Key-chains vs. Key-table Comparison	5
3.1	Organization	5
3.2	Missing Attributes	5
4.	YANG Module	6
5.	Usage	13
6.	Security Considerations	15
7.	IANA Considerations	15
8.	References	15
8.1	Normative References	15
8.2	Informative References	16

[1.](#) Introduction

[RFC7210] defines a standards track key table that is used to store information for routing protocol authentication, including key values, cryptographic algorithms, timing attributes, and their relationships to allow different routing protocols to perform key selection, authentication, and smooth key rollover that ultimately provides routing protocol message security. This document defines a YANG [RFC6020] data model that corresponds to [RFC7210] and enables the use of NETCONF [RFC6241] and YANG to manage routing protocol authentication data.

An earlier version of the key table YANG model [I-D.chen-rtg-key-table-yang] augments from the key-chain YANG model [I-D.acee-rtg-yang-key-chain]. However, because the key-chain YANG model organizes keys in groups, which is different from the key definitions in [RFC7210], this document proposes a new key-table YANG model that is structurally more consistent with the key database defined in [RFC7210].

[1.1.](#) Terminology

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [BCP 14](#), [\[RFC2119\]](#).

[1.2.](#) Tree Diagram

A simplified graphical representation of the data model is presented in [Section 2](#).

Chen

Expires in 6 months

[Page 3]

Internet Draft

Key Table YANG

July 1, 2015

The meaning of the symbols in these diagrams is as follows:

- o Brackets "[" and "]" enclose list keys.
- o Curly braces "{" and "}" contain names of optional features that make the corresponding node conditional.
- o Abbreviations before data node names: "rw" means configuration (read-write), and "ro" state data (read-only).
- o Symbols after data node names: "?" means an optional node and "*" denotes a "list" or "leaf-list".
- o Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (":").
- o Ellipsis ("...") stands for contents of subtrees that are not shown.

[2.](#) Design of the Data Model

This data model is based on the key table defined in [\[RFC7210\]](#). [\[RFC7210\]](#) defines a single table in which the rows represent the individual key entries of the key table. As such, the YANG model consists of one single top-level container named "key-table", in which a YANG list named "security-association-entry" is defined. The entries in the "security-association-entry" YANG list represents the rows that correspond to the cryptographic keys in [\[RFC7210\]](#) key table.

Based on [\[RFC7210\]](#), each "security-association-entry" further consists of mandatory attributes that describe a cryptographic key as well as the key value. In this document, two of these mandatory attributes, "peers" and "protocol-specific-info" are placeholders left for each protocol to define, as prescribed by [\[RFC7210\]](#).

```
module: ietf-key-table
  +--rw key-table
    +--rw security-association-entry* [admin-key-name]
      +--rw admin-key-name          string
      +--rw local-key-name          string
      +--rw peer-key-name           string
      +--rw peers
      +--rw interfaces
        | +--rw (interface-options)
        |   +--:(all-interfaces)
        |   | +--rw all?            empty
        |   +--:(interface-list)
```

Chen

Expires in 6 months

[Page 4]

Internet Draft

Key Table YANG

July 1, 2015

```
    | +--rw interface* if:interface-ref
+--rw protocol          identityref
+--rw protocol-specific-info
+--rw kdf               key-derivation-function-type
+--rw alg-id            cryptographic-algorithm-type
+--rw key               yang:hex-string
+--rw direction        enumeration
+--rw send-lifetime-start lifetime-type
+--rw send-lifetime-end  lifetime-type
+--rw accept-lifetime-start lifetime-type
+--rw accept-lifetime-end lifetime-type
```

[3.](#) Key-chain vs. Key-table Comparison

The key-chain YANG model also proposes a YANG model for key management. This section compares the key-chain model and the key-table model proposed in this document.

[3.1.](#) Organization

The key-chain YANG model groups several keys into a single key chain. It is the key chain that is referenced and applied by routing

protocols. Consequently, the key-chain YANG model defines a hierarchical database, which consists of a top-level key-chain database, and each key-chain database consists of the actual keys that are used by a protocol. A routing protocol that requires encryption or authentication must reference a key-chain instead of the individual keys.

In contrast, [\[RFC7210\]](#) defines a single flat database of keys and their attributes. [\[RFC7210\]](#) does not require that an implementation of key management explicitly group a set of keys into a separate entity that routing protocols reference and use. Consequently, the key-table YANG model defined in this document presents a flat view of the key database.

[3.2.](#) Missing Attributes

The key-chain YANG model defines only a subset of key attributes defined in [\[RFC7210\]](#). Consequently, the key-chain YANG model can only support specific types of deployments requiring authentication. A list of key attributes defined in [\[RFC7210\]](#) and in this document, but not defined in [key-chain], are as follows:

- o Peer key name
- o Interfaces

- o Protocol
- o Key derivation function
- o Direction

In addition to the list above, [key-chain] also does not define attributes that [\[RFC7210\]](#) defined but left the detailed definitions to individual routing protocols. Similar to [\[RFC7210\]](#), this document defines YANG constructs for these attributes and intends for routing protocols to provide the details. The attributes in question are as follows:

- o Peers
- o ProtocolSpecificInfo

4. YANG Module

```
<CODE BEGINS> file "ietf-key-table.yang"

module ietf-key-table {
  namespace "urn:ietf:params:xml:ns:yang:ietf-key-table";
  prefix "keytable";

  import ietf-routing {
    prefix "rt";
  }

  import ietf-interfaces {
    prefix "if";
  }

  import ietf-yang-types {
    prefix "yang";
  }

  organization
    "Ericsson";

  contact
    "I. Chen - ing-wher.chen@ericsson.com";

  description
    "A key table YANG data model based on RFC 7210";
```

Chen

Expires in 6 months

[Page 6]

Internet Draft

Key Table YANG

July 1, 2015

```
revision 2015-06-29 {
  description
    "Revision 2.";
  reference "RFC 7210";
}
```

```
/* Identities */
```

```

identity key-derivation-function {
    description
        "Base identity from which key derivation function " +
        "identities are derived";
}

identity kdf-none {
    base key-derivation-function;
    description
        "This identity represents a cryptographic key that is" +
        "used directly, without a key derivation function.";
}

identity kdf-aes-128-cmac {
    base key-derivation-function;
    description
        "This identity represents the key derivation function that " +
        "uses AES-CMAC using 128-bit keys (RFC 4493).";
}

identity hmac-sha-1 {
    base key-derivation-function;
    description
        "This identity represents the key derivation function that " +
        "uses HMAC using the SHA-1 hash (RFC 2104).";
}

identity cryptographic-algorithm {
    description
        "Base identity from which cryptographic algorithms " +
        "are derived";
}

identity algid-aes-128-cmac {
    base cryptographic-algorithm;
    description
        "This identity represents the cryptographic algorithm " +
        "AES-CMAC using 128-bit keys (RFC 4493).";
}

```

```

identity algid-aes-128-cmac-92 {

```



```

base cryptographic-algorithm;
description
    "This identity represents the cryptographic algorithm " +
    "AES-128-CMAC truncated to 96 bits (RFC 5926).";
}

identity algid-hmac-sha-1-96 {
    base cryptographic-algorithm;
    description
        "This identity represents the cryptographic algorithm " +
        "HMAC SHA-1 truncated to 96 bits (RFC 2104).";
}

identity all-routing-protocols {
    base rt:routing-protocol;
    description
        "All routing protocols";
}

/* Typedefs */

typedef local-key-ref {
    type leafref {
        path "/key-table/security-association-entry/local-key-name";
    }
    description
        "This type is used for leafs within a protocol to reference " +
        "a key using the local-key-name";
}

typedef routing-protocol-type {
    type identityref {
        base rt:routing-protocol;
    }
    description
        "This type identifies the routing protocol";
}

typedef key-derivation-function-type {
    type identityref {
        base key-derivation-function;
    }
    description
        "This type identifies the key derivation function";
}

typedef cryptographic-algorithm-type {

```

```
    type identityref {
      base cryptographic-algorithm;
    }
    description
      "This type identifies the cryptographic algorithm";
  }

typedef lifetime-type {
  type string {
    pattern '\d{4}\d{2}\d{2}\d{2}\d{2}Z';
  }
  description
    "This type identifies a time in the format YYYYMMDDHHSSZ, " +
    "where the first four digits specify the year, " +
    "the next two digits specify the month, " +
    "the next two digits specify the day, " +
    "the next two digits specify the hour, " +
    "the next two digits specify the second, " +
    "ending with the letter 'Z' as a clear indication " +
    "that the time is in Coordinated Universal Time (UTC).";
}

/* Groupings */

grouping protocol-authentication-key-grouping {
  description
    "A grouping to be used by routing protocols to specify a key.";
  container authentication {
    description
      "Key selection criteria for authentication.";
    container out-key {
      description
        "Specify the key selection criteria for " +
        "outbound traffic as defined in RFC 7210 Section 3.";
      leaf peer-identifier {
        type string;
        description
          "Specify a peer.";
      }
    }
  }
  container in-key {
    description
      "Specify the key selection criteria for " +
      "inbound traffic as defined in RFC 7210 Section 3.";
    leaf key-identifier {
      type local-key-ref;
    }
  }
}
```

```
description
    "Specify the local key.";
```

```
    }
    leaf peer-identifier {
        type string;
        description
            "Specify a peer.";
    }
}
}
```

```
/* The key-table model */
```

```
container key-table {
    description
        "The key table of all managed cryptographic keys " +
        "of a device.";
    list security-association-entry {
        key "admin-key-name";
        description
            "A key table entry that specifies a key " +
            "and its attributes.";
        leaf admin-key-name {
            type string;
            description
                "A human-readable string that identifies the key";
        }
        leaf local-key-name {
            type string;
            mandatory true;
            description
                "A string for a routing protocol to identify the key " +
                "in the key-table when received in a message.";
        }
        leaf peer-key-name {
            type string;
            mandatory true;
            description
                "A string to identify the key to be used by " +
                "the local system in an outgoing message.";
```

```

}
container peers {
  description
    "Specify the peer systems that also have this key " +
    "in their database. The format of this field is " +
    "left to protocols to define.";
}
container interfaces {
  description

```

Chen

Expires in 6 months

[Page 10]

Internet Draft

Key Table YANG

July 1, 2015

```

    "Specify the interfaces to which they key may be applied.";
  choice interface-options {
    mandatory true;
    description
      "The option to apply this key to all interfaces or " +
      "to a pre-defined list of interfaces.";
    case all-interfaces {
      leaf all {
        type empty;
        description
          "This key applies to all interfaces.";
      }
    }
    case interface-list {
      leaf-list interface {
        type if:interface-ref;
        description
          "This key applies to the identified interfaces.";
      }
    }
  }
}
leaf protocol {
  type identityref {
    base rt:routing-protocol;
  }
  mandatory true;
  description
    "Specify a single routing protocol where this key " +
    "may be used to provide cryptographic protection.";
}
container protocol-specific-info {

```

```

    description
        "This field contains protocol-specified information " +
        "that maybe useful for a protocol to apply the key " +
        "correctly. This field is left for each protocol " +
        "to define.";
}
leaf kdf {
    type key-derivation-function-type;
    mandatory true;
    description
        "Specify the key derivation function to be used " +
        "with this key.";
}
leaf alg-id {
    type cryptographic-algorithm-type;
    mandatory true;

```

Chen

Expires in 6 months

[Page 11]

Internet Draft

Key Table YANG

July 1, 2015

```

    description
        "Specify the cryptographic algorithm to be used" +
        "with this key.";
}
leaf key {
    type yang:hex-string;
    mandatory true;
    description
        "The long-lived symmetric cryptographic key.";
}
leaf direction {
    type enumeration {
        enum "in" {
            description
                "The key is used for inbound traffic.";
        }
        enum "out" {
            description
                "The key is used for outbound traffic.";
        }
        enum "both" {
            description
                "The key is used for both inbound and outbound " +
                "traffic.";
        }
    }
}

```

```

        enum "disabled" {
            description
                "The key is disabled and cannot be used for " +
                "either inbound or outbound traffic.";
        }
    }
    mandatory true;
    description
        "The value of the direction must be one of 'in', 'out', " +
        "'both', and 'disabled'. The actual allowed value " +
        "is left for routing protocols to define.";
}
leaf send-lifetime-start {
    when "../direction = 'out' or 'both'" {
        description
            "Only valid if a key can be used in outbound traffic.";
    }
    type lifetime-type;
    mandatory true;
    description
        "Specify the earliest date and time at which this key " +
        "should be considered for use when sending traffic.";
}

```

```

leaf send-lifetime-end {
    when "../direction = 'out' or 'both'" {
        description
            "Only valid if a key can be used in outbound traffic.";
    }
    type lifetime-type;
    mandatory true;
    description
        "Specify the latest date and time at which this key " +
        "should be considered for use when sending traffic.";
}
leaf accept-lifetime-start {
    when "../direction = 'in' or 'both'" {
        description
            "Only valid if a key can be used by inbound traffic.";
    }
    type lifetime-type;
    mandatory true;
}

```

```

        description
            "Specify the earliest date and time at which this key " +
            "should be considered for processing received traffic.";
    }
    leaf accept-lifetime-end {
        when "../direction = 'in' or 'both'" {
            description
                "Only valid if a key can be used by inbound traffic.";
        }
        type lifetime-type;
        mandatory true;
        description
            "Specify the latest date and time at which this key " +
            "should be considered for processing received traffic.";
    }
}
}
}
}

```

<CODE ENDS>

5. Usage

A routing protocol that requires a cryptographic key would reference a YANG grouping called "protocol-authentication-key-grouping" that is defined in ietf-key-table YANG model. "protocol-authentication-key-grouping" includes the key selection criteria described in [Section 3 in \[RFC7210\]](#). In its simplest form, a protocol only needs to identify the local-key-name, similar to the key-chain references used in existing OSPF [[I-D.ospf-yang](#)] and ISIS [[I-D.isis-yang-isis-cfg](#)] routing protocol YANG models.

Chen

Expires in 6 months

[Page 13]

Internet Draft

Key Table YANG

July 1, 2015

```

module foo-routing-protocol {
    namespace "http://www.example.com/foo-routing-protocol";
    prefix "foo-routing";

    import ietf-routing {
        prefix "rt";
    }

    import ietf-interfaces {
        prefix "if";
    }
}

```

```

import ietf-key-table {
    prefix "keytable";
}

organization "TBD";
contact "TBD";
description
    "A sample routing protocol that uses ietf-key-table for " +
    "authentication.";

revision 2015-06-29 {
    description
        "Initial revision.";
    reference "";
}

identity foo-routing-protocol {
    base "rt:routing-protocol";
    description
        "foo-routing-protocol";
}

augment "/rt:routing/rt:routing-instance/rt:routing-protocols" +
    "/rt:routing-protocol" {
    when "rt:type = foo-routing-protocol" {
        description
            "This augment is only valid for a routing protocol " +
            "instance of foo-routing-protocol";
    }
    description
        "foo-routing-protocol augmentation.";
    container foo {
        description
            "foo-routing-protocol";
        list interface {
            key "name";

```

```

description
    "List of foo-routing-protocol interfaces.";
leaf name {
    type if:interface-ref;

```



```

        description
            "Specify an interface.";
    }
    uses keytable:protocol-authentication-key;
}
}
}
}
}

```

6. Security Consideration.

TBD.

7. IANA Considerations

TBD.

8. References

8.1. Normative References

- [RFC7210] Housley, R., Polk, T., Hartman, S., and D. Zhang, "Database of Long-Lived Symmetric Cryptographic Keys", [RFC 7210](http://www.rfc-editor.org/info/rfc7210), April 2014, <<http://www.rfc-editor.org/info/rfc7210>>.
- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", [RFC 6020](http://www.rfc-editor.org/info/rfc6020), DOI 10.17487/RFC6020, October 2010, <<http://www.rfc-editor.org/info/rfc6020>>.
- Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", [RFC 6241](http://www.rfc-editor.org/info/rfc6241), DOI 10.17487/RFC6241, June 2011, <<http://www.rfc-editor.org/info/rfc6241>>.
- [I-D.acee-rtg-yang-key-chain] Lindem, A., Qu, Y., Yeung, D., Chen, I., Zhang, J., and Y. Yang, "Key Chain YANG Data Model", [draft-acee-rtg-yang-key-chain-04](http://www.rfc-editor.org/draft-acee-rtg-yang-key-chain-04) (work in progress), April 2015.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](http://www.rfc-editor.org/info/bcp14), [RFC 2119](http://www.rfc-editor.org/info/rfc2119), March 1997.

[8.2.](#) Informative References

[I-D.chen-rtg-key-table-yang] Chen, I., "YANG Data Model for [RFC 7210](#) Key Table", [draft-chen-rtg-yang-key-table-00](#) (work-in-progress), March 2015.

[I-D.ospf-yang] Yeung, D., Qu, Y., Zhang, J., Bogdanovic, D., and K. Sreenivasa, "[draft-ietf-ospf-yang-00](#) (work in progress)", March 2015.

[I-D.isis-yang-isis-cfg] Litkowski, S., Yeung, D., Lindem, A., Zhang, J., and L. Lhotka, "[draft-ietf-isis-yang-isis-cfg-03](#) (work in progress)", June 2015.

Author's Address

I. Chen
Ericsson
Email: ing-wher.chen@ericsson.com

Chen

Expires in 6 months

[Page 16]