

SIDR Working Group  
Internet-Draft  
Intended status: BCP  
Expires: March 16, 2014

S. Turner  
IECA, Inc.  
K. Patel  
Cisco Systems  
R. Bush  
Internet Initiative Japan, Inc.  
September 12, 2013

Router Keying for BGPsec  
draft-ietf-sidr-rtr-keying-02

Abstract

BGPsec-speaking routers must be provisioned with private keys and the corresponding public key must be published in the global RPKI (Resource Public Key Infrastructure). This document describes two ways of provisioning public/private keys, router-driven and operator-driven.

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 <http://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 August 27, 2013.

Copyright Notice

Copyright (c) 2013 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

Internet-Draft

Router Keying for BGPsec

September 12, 2013

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.

## 1. Introduction

BGPsec-speaking routers must be provisioned with private keys and the corresponding public key must be published in the global RPKI (Resource Public Key Infrastructure). The public key is published in the RPKI in the form of a certificate [[I-D.ietf-sidr-bgpsec-pki-profiles](#)]. This document describes two methods for generating the necessary public/private key-pair: router-driven and operator-driven.

The difference between the two methods is where the keys are generated: on the router in the router-driven method and elsewhere in the operator-driven model. Different equipment necessitates the two methods. Some equipment doesn't allow the private key to be off-loaded while other equipment does. Off-loading private keys supports hot-swappable routers that need to have the same private key needs installed in the soon-to-be online router that was installed in the soon-to-be offline router.

The remainder of this document describes how operators can use the two methods to provision new and existing routers.

Note that in both models, the key pair is for algorithms defined in [[I-D.ietf-sidr-bgpsec-algs](#)]. The first version specifies ECDSA on the P-256 curve.

## 2. Terminology

The key words "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 [RFC 2119](#) [[RFC2119](#)].

It is assumed that the reader understands BGPsec [[I-D.ietf-sidr-bgpsec-overview](#)] [[I-D.ietf-sidr-bgpsec-protocol](#)], the RPKI [[RFC6480](#)], and [[I-D.ietf-sidr-bgpsec-pki-profiles](#)].

## 3. Provisioning a New Router

When commissioning a new router, operators may use either the router-

driven or operator-driven methods. Regardless of the method chosen, the operator first needs to establish a secure communication channel with the router. Operators use the router-specific procedures to enable them to connect to the router via an SSH session.

The SSH encryption, integrity, authentication, and key exchange mechanisms used by the router and operator SHOULD be of comparable strength to BGPSEC key, which is 128-bit strength, e.g., for encryption: aes128-cbc [[RFC4253](#)] and AEAD\_AES\_128\_GCM [[RFC5647](#)], for integrity: hmac-sha2-256 [[RFC6668](#)] and AESAD\_AES\_128\_GCM [[RFC5647](#)], for authentication: ecdsa-sha2-nistp256 [[RFC5656](#)], and for key exchange: ecdh-sha2-nistp256 [[RFC5656](#)].

Note that if the router supports public key certificates at this point, which would have had to have been provided by the operator at this point, x509v3-ecdsa-sha2-nistp256 [[RFC6187](#)] could be used for authentication. The SSH certificate, profiled in [[RFC6187](#)], would be different than the BGPSEC certificate.

### [3.1.](#) Router-Generated Keys

In the router-driven method, once an SSH session is established between the operator and the router the operator issues a command, or commands, to generate the public/private key pair on the router, to generate the PKCS#10 request that includes the router number and public key, and to sign the PKCS#10 with the private key. [I-D.ietf-sidr-bgpsec-pki-profiles] specifies the format for the PKCS #10 request and the algorithm used to generate the signature is specified in [[I-D.ietf-sidr-bgpsec-algs](#)].

The PKCS#10 request, which includes the public key the router wants certified, can be directly transferred to the RPKI CA over the Ethernet port if the router supports protocols such as FTP and HTTP [[RFC2585](#)] using the application/pkcs10 media type [[RFC5967](#)] or EST (Enrollment over Secure Transport) [[I-D.ietf-pkix-est](#)]. The CA returns a successful request as a PKCS#7 [I-D.ietf-sidr-bgpsec-pki-profiles], which includes the certificate, and uploads the certificate to the global RPKI. The response can be returned using the application/pkcs7-mime media type [[RFC5751](#)] if the router supports protocols such as FTP and HTTP.

The PKCS#10 request can also be indirectly transferred to the RPKI CA through the operator. The operator off-loads the PKCS#10 and uploads the request to their RPKI software management tools. The tools create and publish the certificate for the public key, and return the PKCS#7 to the router.

The router SHOULD extract the certificate from the PCKCS#7 and verify that the private key corresponds to the returned public key. The router SHOULD inform the operator that it has successfully received its certificate; this mechanism is out of scope. When the keys do not correspond, the router SHOULD inform the operator; this mechanism is out of scope. The router SHOULD also verify the returned

certificate back to a trust anchor, but to perform this verification either the CA's certificate needs to be installed on the router via the CLI or the CA's certificate needs to be returned along with the router's certificate in the PKCS#7. The router SHOULD inform the operator if the signature does not validate to a trust anchor; this notification mechanism is out of scope. After performing these checks, the router need not retain the certificate.

Note that even if the operator can not get the private key off the router this signature still provides a linkage between a private key and a router. That is the server can verify the proof of possession (POP), as required by [\[RFC6484\]](#).

### [3.2.](#) Operator-Generated Keys

In the operator-driven method, the operator generates the private key and it is installed over the SSH connection established between the operator and the router. Note that cut/copy and paste operations for keys over a certain sizes is error-prone.

The operator uses their RPKI management tools to generate the keys, the PKCS#10 certification request, the certificate, and the PKCS#7 certification response as well as publish the certificate for the public key in the global RPKI. The private key MUST support the algorithm specified in [\[I-D.ietf-sidr-bgpsec-algs\]](#), which for ECDSA is specified in [\[RFC5915\]](#). The PKCS#10 and PKCS#7 are as specified in [\[I-D.ietf-sidr-bgpsec-pki-profiles\]](#).

Along with the PKCS#7, the operator returns the private key. The

private key is encapsulated in a PKCS #8 [[RFC5958](#)], the PKCS#8 is further encapsulated in a CMS (Cryptographic Message Syntax) SignedData [[RFC5652](#)], and signed by the operator's EE certificate.

The router SHOULD verify the signature on the encapsulated PKCS#8 to ensure the returned private key in fact came from the operator, but this requires that the operator also provision via the CLI or include in the SignedData the RPKI CA certificates and operator's EE certificates. The router SHOULD inform the operator if the signature does not validate to a trust anchor; this notification mechanism is out of scope.

The router SHOULD extract the certificate for the PKCS#7 and verify that the private key corresponds to the returned public key. The router SHOULD inform the operator that it has successfully received its certificate; this mechanism is out of scope. When the keys do not correspond, the router SHOULD inform the operator; this mechanism is out of scope. The router SHOULD also verify the returned certificate back to a trust anchor, but to perform this verification

either the CA's certificate needs to be installed on the router via the CLI or the CA's certificate needs to be returned along with the router's certificate in the PKCS#7. The router SHOULD inform the operator if the signature does not validate to a trust anchor; this notification mechanism is out of scope. After performing these checks, the router need not retain the certificate.

## [5.](#) Other Use Cases

Current router code generates private keys for uses such as SSH, but the private keys may not be seen or off-loaded via CLI or any other means. While this is good security, it creates difficulties when a routing engine or whole router must be replaced in the field and all software which accesses the router must be updated with the new keys. Also, the initial contact with a new routing engine requires trust in the public key presented on first contact.

To allow operators to quickly replace routers without requiring update and distribution of the corresponding public keys in the RPKI, routers SHOULD allow the private BGPsec key to be off-loaded via the CLI, NetConf (see [[RFC6470](#)]), SNMP, etc. This lets the operator upload the old private key via the mechanism used for operator-

generated keys, see [Section 3.2](#).

## [6.](#) Security Considerations

Operator-generated keys could be intercepted in transport and the recipient router would have no way of knowing a substitution had been made or that the key had been disclosed by a monkey in the middle. Hence transport security is strongly RECOMMENDED. As noted in [Section 3](#), the level of security provided by the transport security SHOULD be commensurate with the BGPsec key. Additionally, operators SHOULD ensure the transport security implementation is up to date and addresses all known implementation bugs.

All generated key pairs MUST be generated from a good source of non-deterministic random input [[RFC4086](#)] and the private key MUST be protected in a secure fashion. Disclosure of the private key leads to masquerade [[RFC4949](#)]. The local storage format for the private key is a local matter.

Though the CA's certificate is installed on the router and used to verify the returned certificate is in fact signed by the CA, the revocation status of the CA's certificate is not checked. The operator MUST ensure that installed CA certificate is valid.

Operators need to manage their SSH keys to ensure only those authorized to access the router can. As employees no longer need

access to the router, their keys SHOULD be removed from the router.

## [7.](#) IANA Considerations

This document has no IANA Considerations.

## [8.](#) References

### [8.1.](#) Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

[RFC4086] Eastlake 3rd, D., Schiller, J., and S. Crocker, "Randomness Requirements for Security", [BCP 106](#), [RFC 4086](#),

June 2005.

- [RFC4253] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Transport Layer Protocol", [RFC 4253](#), January 2006.
- [RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, [RFC 5652](#), September 2009.
- [RFC5915] Turner, S. and D. Brown, "Elliptic Curve Private Key Structure", [RFC 5915](#), June 2010.
- [RFC5958] Turner, S., "Asymmetric Key Packages", [RFC 5958](#), August 2010.
- [I-D.ietf-sidr-bgpsec-algs]  
Turner, S., "BGP Algorithms, Key Formats, & Signature Formats", [draft-ietf-sidr-bgpsec-algs](#) (work in progress), March 2013.
- [I-D.ietf-sidr-bgpsec-pki-profiles]  
Reynolds, M., Turner, S., and S. Kent, "A Profile for BGPSEC Router Certificates, Certificate Revocation Lists, and Certification Requests", [draft-ietf-sidr-bgpsec-pki-profiles](#) (work in progress), April 2013.

## [8.2.](#) Informative References

- [I-D.ietf-sidr-bgpsec-overview]  
Lepinski, M. and S. Turner, "An Overview of BGPSEC", [draft-ietf-sidr-bgpsec-overview](#) (work in progress), July 2013.

Turner, et al.

Expires March 16, 2014

[Page 6]

---

Internet-Draft

Router Keying for BGPsec

September 12, 2013

- [I-D.ietf-sidr-bgpsec-protocol]  
Lepinski, M., "BGPSEC Protocol Specification", [draft-ietf-sidr-bgpsec-protocol](#) (work in progress), February 2013.
- [I-D.ietf-pkix-est]  
Pritikin, M, Yee, P., and D. Harkins "Enrollment over Secure Transport", [draft-ietf-pkix-est](#) (work in progress),

August 2013.

- [IEEE-802.3]  
ISO/IEC 8802-3 Information technology -  
Telecommunications and information exchange between  
systems - Local and metropolitan area networks -  
Common specifications - Part 3: Carrier Sense  
Multiple Access with Collision Detection (CSMA/CD)  
Access Method and Physical Layer Specifications, 2008.
- [RFC2585] Housley, R. and P. Hoffman, "Internet X.509 Public Key  
Infrastructure Operational Protocols: FTP and HTTP",  
[RFC 2585](#), May 1999.
- [RFC4253] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH)  
Transport Layer Protocol", [RFC 4253](#), January 2006.
- [RFC4949] Shirey, R., "Internet Security Glossary, Version 2", FYI  
36, [RFC 4949](#), August 2007.
- [RFC5647] Igoe, K. and J. Solinas, "AES Galois Counter Mode for the  
Secure Shell Transport Layer Protocol", [RFC 5647](#), August  
2009.
- [RFC5656] Stebila, D. and J. Green, "Elliptic Curve Algorithm  
Integration in the Secure Shell Transport Layer",  
[RFC 5656](#), December 2009.
- [RFC5751] Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet  
Mail Extensions (S/MIME) Version 3.2 Message  
Specification", [RFC 5751](#), January 2010.
- [RFC5967] Turner, S., "The application/pkcs10 Media Type", [RFC 5967](#),  
August 2010.
- [RFC6187] Igoe, K. and D. Stebila, "X.509v3 Certificates for Secure  
Shell Authentication", [RFC 6187](#), March 2011.
- [RFC6470] Bierman, A., "Network Configuration Protocol (NETCONF)  
Base Notifications", [RFC 6470](#), February 2012.

Turner, et al.

Expires March 16, 2014

[Page 7]

---

Internet-Draft

Router Keying for BGPsec

September 12, 2013



- [RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", [RFC 6480](#), February 2012.
- [RFC6668] Bider, D. and M. Baushke, "SHA-2 Data Integrity Verification for the Secure Shell (SSH) Transport Layer Protocol", [RFC 6668](#), July 2012.

Authors' Addresses

Sean Turner  
IECA, Inc.  
3057 Nutley Street, Suite 106  
Fairfax, Virginia 22031  
US

Email: [turners@ieca.com](mailto:turners@ieca.com)

Keyur Patel  
Cisco Systems  
170 West Tasman Drive  
San Jose, CA 95134  
US

Email: [keyupate@cisco.com](mailto:keyupate@cisco.com)

Randy Bush  
Internet Initiative Japan, Inc.  
5147 Crystal Springs  
Bainbridge Island, Washington 98110  
US

Phone: +1 206 780 0431 x1

Email: [randy@psg.com](mailto:randy@psg.com)

