I2NSF
Internet-Draft
Intended status: Informational
Expires: January 23, 2020

Y. Nir
DellEMC

July 22, 2019

A Data Center Profile for Software Defined Networking (SDN)-based IPsec

draft-nir-i2nsf-ipsec-dc-prof-00

Abstract

This document presents two profiles for configuring IPsec within a data center using an SDN controller and the YANG model described in the sdn-ipsec draft.

Two profiles are described to allow both the IKE and IKE-less cases because some data centers may be required to use a standardized method of key exchange rather than SDN.

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 January 23, 2020.

1. Introduction

[sdn-ipsec] describes a YANG model that allows a software defined networking (SDN) controller to configure the use of IP security (IPsec - [RFC4301]) and optionally the Internet Key Exchange protocol (IKEv2 - [RFC7296]) to secure IP traffic between the hosts that it controls.

The SDN-IPsec document allows for configuration of most of the options available in IPsec. However, not every one of those options are appropriate for all use cases.

The use case that is covered here is the need to encrypt traffic between hosts within a data center. As explained in Section 2, data centers cannot be considered a secure environment where internal communications are safe behind the firewall. One way to protect the internal traffic is to configure TLS pair-wise between the hosts, but [sdn-ipsec] provides a more convenient, automated solution.

This document presents two profiles that are appropriate for encrypting traffic among the hosts in a data center, one with and one without the use of IKE.

1.1. Terminology

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

"Security Controller" or "SC" is an SDN controller used to configure security policy. For the purposes of this document, we limit the use of this term to an SDN controller that distributes IPsec policy.

"Data center hosts" is the term we use for any machine in the data center that communicates using Internet Protocol (IP) with other machines, both within and outside the data center.

"Network Security Functions" or NSF is the term used for a host in

the data plane that implements a security function. For the purposes of this document we will call a host that has an IPsec stack and the software necessary to be configured by an SC an "IPsec NSF".

"Control Domain" will be used here to mean the set of all IPsec NSFs controlled by a particular security controller. The controller can set up security associations within the control domain, but any associations from within the domain to hosts or gateways outside of the domain have to be configured on the remote host as well. The controller can, however, configure the local side of things, as mentioned in Section 3.4.

2. Assumptions About The Evnironment

A modern data center usually has many systems from several different vendors containing data with varying levels of sensitivity. In the past people often assumed that the data center was protected from traffic interception by physical security. It was assumed that traffic within the data center or within the corporate network could safely be sent in the clear. This perception no longer holds if it ever did. (TODO: citation needed).

The servers in today's data center are connected both to corporate systems outside the data center as well as to public clouds and the Internet. Even if physical security is maintained, the threat of a compromised server intercepting internal traffic is very real. In practice, even the physical security cannot be consistently maintained, as technicians from multiple vendors are often allowed physical access to the data center and supervision of those technicians is often lax.

Additionally, certain industries or types of data are regulated to require encryption of all data in transit. Medical information, personal information, and financial information are examples of data that often requires protection both at rest and in transit. For these reasons it is often necessary to encrypt traffic within the data center, between the data center and corporate networks, and between the data center and public clouds.

IPsec is a good option for encrypting traffic between servers. It

provides the required confidentiality and message authentication, and it is included in every common operating system as well as third party vendor products It can be imposed by server administrators without any changes to or configuration of the applications running on the servers.

The problem with using IPsec has been that its configuration is difficult. The data structures described in [RFC4301] require that data center hosts should all be configured with Peer Authentication Database (PAD) and Security Policy Database (SPD) entries for each peer host, plus either pair-wise shared secrets or public-key based credentials. There has never been a scalable way to perform this

mass configuration until [sdn-ipsec], which allows an SDN controller (renamed to Security Controller) to configure all of the necessary information so that any two data center hosts can communicate using IPsec.

The following assumptions are made:

- That an SC as described in [sdn-ipsec] is present in the data center.
- That the data center hosts are also IPsec NSFs, that they implement the NSF role of an [sdn-ipsec] implementation, and that they can all be configured by the above-mentioned SC.
- That the IPsec NSFs are relatively new, so that they include implementations of current cryptographic algorithms.
- That the connection between the SC and the IPsec NSFs is secure. Specifically, that it is safe to transmit keys and secret credentials over that connection.
- That the SC can produce enough good random bits to periodically produce pair-wise keys for as many IPsec NSFs as it can control.
- That both the IKE and IKE-less cases from [sdn-ipsec] are technically viable. In other words, the software on the IPsec NSFs can accommodate both.

If the last point does not hold, and the NSFs can only accommodate

IPsec (but not IKE), then only the IKE-less option is viable. At this point in time, I am not aware of any such NSFs.

2.1. Block and Bypass Traffic

Not all nodes in a data center are supposed to communicate with one another at all, and some traffic does not need to be encrypted. In other words, a mesh is not the most appropriate topology for IPsec on the network. The controller can enforce this lack of communication with policy that blocks all communications that are not needed with the action "block" and allow some unprotected traffic with the action "bypass".

This means that with N IPsec NSFs, there will be far less than N^2 security associations. A mesh is still a valid configuration, but it's not usually the most appropriate. Using "block" actions to prevent unwanted communications is as much a part of enforcing a security policy in the data center as encrypting legitimate traffic.

The particulars of what traffic should be allowed in the clear, what should be protected, and what should be blocked are, of course, unique to each organization and this document cannot make any rules about that. Most often, the last or "cleanup" rule in the policy should be a universal "block" rule.

3. Profiles For Data Center Hosts

This section presents two profiles for using IPsec in the data center: one that includes IKE, and one that does not. The choice between these two is entirely up to the regulatory regime. The IKE-less profile is simpler and requires less components. It is preferable unless the regulatory regime demands the use of an Authenticated Key Exchange (AKE) method such as IKEv2.

3.1. IKE Profile

With an IKE profile, all pairs of hosts that are supposed to communicate securely with one another SHALL be issued shared secrets. The shared secrets MUST be generated independently of one another by the controller using a true random number generator (TRNG) or a secure pseudo-random number generator (PRNG) for each pair of hosts.

Only IKEv2 will be used.

The identities configured for the PAD can either be meaningful names from the configuration of the controller, or they can be generated sequentially by the controller. In either case the ID type SHALL be Key ID (See section 4.4.3.1 of [RFC4301])

The algorithms used within IKEv2 SHALL be selected from among those marked MUST, SHOULD, and SHOULD+ in [RFC8247] without the "(IoT)" label, or a newer RFC that will obsolete RFC 8247 (TODO: why is this not a BCP?)

The lifetime for an IKE SA SHALL be 24 hours. The lifetime for an ESP SA SHALL be 8 hours.

For both IKE and IPsec, the controller MUST specify exactly one set of algorithms for each pair of nodes. The controller SHOULD specify one set of algorithms for all the associations in the system, unless one of the following applies:

1. The preferred algorithm is not supported by all nodes, so those that do not support it have to use another algorithm.

2. Different algorithms have different performance on different NSFs. For example, AES-GCM is faster than ChaCha20-Poly1305 on

Intel platforms, while ChaCha20-Poly1305 is faster on ARM platforms. It can be advantageous to use one or the other depending on the types of systems communicating.

The rest of the properties are similar to those in the IKE-less profile (Section 3.2)

3.2. IKE-Less Profile

All security associations MUST have selectors (see section 4.4.1.1 of [RFC4301]) that have a single local address and a single remote address with no value for protocols or ports. TBD: exception for multi-homed hosts?

All security associations are provided proactively. The controller

does not wait for a request from the NSF for an SA.

The controller SHOULD refresh the SAs every hour, and MAY do this more often if the volume of traffic exceeds the limits of the algorithms used.

3.3. Propertied Common to Both Profile

All SPD entries MUST have selectors that have a single local address and a single remote address with no value for protocols or ports. This, in the IKE case will lead to SAs as described in the first paragraph of Section 3.2, so this requirement does not need to be repeated. TBD: exception for multi-homed hosts?

All algorithms used in IPsec MUST be those marked MUST, SHOULD, and SHOULD+ in [RFC8221], or a newer RFC that will obsolete RFC 8221 (TODO: why is this not a BCP?)

All SAD entries MUST be regular ESP [RFC4303]. AH [RFC4302] and WESP [RFC5840] are not supported in this profile.

All SAs SHOULD use tunnel-mode. They MAY use transport mode only if all NSFs support this.

3.4. Communications Outside the Domain

Associations between NSFs in the domain and NSFs that are not in the domain are outside the scope of this document. The security controller may configure the NSFs in the domain with the IKE case, but success of the communications depends on the other NSF being configured in a compatible way.

Because of this dependency, the advice in this document do not apply: It is fine to support multiple algorithms, it is fine to support subnets and/or specific protocols and ports, and it is also OK to use other ID types and certificates. That configuration can co-exist in the NSFs with the configuration specified in this profile, but is out of scope here.

4. Rationale for the Properties in the Profile

The sub-sections below explain the rationale for the content of Section 3.

4.1. Why IKE-less is preferrable

IKEv2 [RFC7296] is a protocol for authenticating peers and generating traffic encryption keys. It allows peers that have a good random number generator to be configured either once or rarely, and still be able to communicate securely over the Internet.

IKEv2 thus addresses two issues that are not at all problematic for IPsec NSFs that are configured by an SC. The SC can configure the NSF as often as necessary, and already has the identities established through its own secure channel with those NSFs.

For this reason, setting up the traffic keys directly by the SC where it exists and controls all the relevant hosts is not an inferior solution. It should be preferred for its simplicity, its lower latency, and because it avoids relying on the random number generator within the NSF.

4.2. Shared Secrets vs PKI

We chose to use shared secrets in Section 3.1 because they are simpler than PKI and require less infrastructure. PKI has an advantage when configuring the hosts pair-wise is difficult. However, using a security controller means that changing the configuration or generating pair-wise secrets for even a large number of hosts is attainable. With this change of assumption, it no longer makes sense to use PKI with its expiration times, revocation checks and hierarchical signature verification.

4.3. Why just one algorithm in IKE

IKEv2 allows peers to each support multiple algorithms, and the protocols selects one that is supported by both. This is a good feature for interoperability between peers that are configured separately. When configuring the peers with SDN IPsec, both peers

are configured by the same controller, so there is no reason for them

to offer any algorithm except the one preferred by the controller.

4.4. Why not MUST-

In both Section 3.1 and Section 3.3 we required the use of algorithms marked as MUST, SHOULD, or SHOULD+. We excluded those marked as MUST-, even though these seem to be at a higher level of preference than those marked SHOULD or SHOULD+.

The reason for this is that despite what [RFC8221] says, algorithms tend to be deprecated quickly and may fall from MUST- to MAY or even MUST NOT. The only algorithm marked as MUST- in those drafts in HMAC-SHA1, and it would have been at the MAY or lower level had it not been for the fact that it is the most widely deployed algorithm, and disabling it may lead to interoperability problems.

In a new deployment such as this, there is no reason to keep using such an outdated algorithm that is very likely on its way out.

4.5. Proactive vs Reactive Model

The profile in Section 3.2 is proactive. SAs are installed in the NSFs along with the policy, and are maintained as long as the policy remains. We never wait for the NSF to request an SA. There are two reasons for this:

1. Creating the SAs proactively eliminates any latency in processing a packet at the NSF.

2. The cost of an unused SA is very low in the NSF - usually on the order of a few hundreds of bytes. The cost at the controller of managing these SAs is also low. If SAs are generated every 8 hours and there are 1000 IPsec NSFs in a mesh, that's still just a million tunnels and only 35 needing to be rekeyed per second.

5. IANA Considerations

There are no requests for IANA in this document.

6. Security Considerations

The entire document is about security. The considerations in [sdn-ipsec] apply. Additionally, Section 4 contains explanation of the thinking behind the security decisions in this document.

The environment where this profile is expected to be used is described in the Introduction (Section 1), and is an internal network

of a data center rather than the open Internet. Despite this, no assumptions are made about the network between IPsec NSFs being in any way safer than the open Internet: the connection between controller and NSF is required to be secure, and traffic keys are set up in a secure way: either over the controller-NSF secure connection, or using IKEv2.

The communication channel between the security controller and the NSF is required to be secure because it carries traffic keys, credentials, or both.

A risk that is not addressed in this document is that of an attacker blocking or delaying messages from the controller to the NSFs so as to prevent the timely setup of security associations. Such an attack can lead to denial of service if the IPsec NSFs are configured to fail closed, or to sending traffic in the clear if they are configured to fail open, which may be valid if it is expected that only some of the traffic in the data center is to be encrypted. This risk has to be mitigated by normal data center operations which should ensure that nodes in the data center, in this case the controller and the NSF, are not blocked.

7. ToDo

Need to add a reference to https://csrc.nist.gov/publications/detail/sp/800-77/rev-1/draft

8. References

8.1. 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>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8221] Wouters, P., Migault, D., Mattsson, J., Nir, Y., and T. Kivinen, "Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)", RFC 8221, DOI 10.17487/RFC8221, October 2017, <https://www.rfc-editor.org/info/rfc8221>.

[RFC8247] Nir, Y., Kivinen, T., Wouters, P., and D. Migault, "Algorithm Implementation Requirements and Usage Guidance for the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 8247, DOI 10.17487/RFC8247, September 2017, <https://www.rfc-editor.org/info/rfc8247>.

[sdn-ipsec] Lopez, R., Lopez-Millan, G., and F. Pereniguez-Garcia, "Software-Defined Networking (SDN)-based IPsec Flow Protection", draft-ietf-i2nsf-sdn-ipsec-flow-protection-04 (work in progress), March 2019.

8.2. Informative References

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/info/rfc4301>.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI 10.17487/RFC4302, December 2005, <https://www.rfc-editor.org/info/rfc4302>.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, DOI 10.17487/RFC4303, December 2005, <https://www.rfc-editor.org/info/rfc4303>.

[RFC5840] Grewal, K., Montenegro, G., and M. Bhatia, "Wrapped Encapsulating Security Payload (ESP) for Traffic Visibility", RFC 5840, DOI 10.17487/RFC5840, April 2010, <https://www.rfc-editor.org/info/rfc5840>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

Author's Address

Yoav Nir

DellEMC
9 Andrei Sakharov St
Haifa 3190500
Israel

Email: ynir.ietf@gmail.com