General Intelligence and
Security Service
*Ministry of the Interior and
Kingdom Relations*

Deployment Advisory OpenVPN-NL version 2.5

Date February 13, 2023

Colofon

Our reference number	:	946e0fa0-or1-1.38
	:	T +31 79 320 50 50
	:	F +31 70 320 07 33
	:	P.O. Box 20010 2500 EA The Hague The Netherlands
Copy number	:	
Author(s)	:	
Number of enclosures	:	0

Table of contents

Version history

Document version	*Date*	*Comments / changes*
0.7	27 Oct 2022	First draft
0.8	22 Dec 2022	Internal review
0.9	20 Jan 2023	Further internal review: merge technical appendices into this document; *major* update to "Appendix 1 - OpenVPN-NL options"
1.0	22 Feb 2023	Final version

The current document is the NLNCSA Deployment Advisory for OpenVPN-NL 2.5.x. This Deployment Advisory updates the prior Deployment Advisory v1.4 with additional information and requirements, based on NLNCSA's most up-to-date expertise and information.

For users still deploying OpenVPN-NL 2.4.x (which is "deprecated", as defined in paragraph "Lifecycle"), NLNCSA *recommends* following the recommendations and (partly new) requirements in this Deployment Advisory.

However, deployments of OpenVPN-NL 2.4.x under the prior deployment advisory v1.4 (document identifier 8f495785-or1-1.4) will continue to be supported by NLNCSA (like any other "deprecated" configuration), until the last release of OpenVPN-NL 2.4.x becomes "untrusted/insecure" (as defined in paragraph "Lifecycle").

Management Summary

The open source VPN (virtual private network) implementation OpenVPN has been hardened and documented by Fox-IT under direction of the NLNCSA (Netherlands National Communications Security Agency, in Dutch “NBV”: Nationaal Bureau voor Verbindingsbeveiliging) and has been successfully evaluated to NLNCSA Evaluation criteria 2020 up to Dep. V. (RESTRICTED). Since then Fox-IT has maintained OpenVPN-NL under direction of the NLNCSA. This report describes the outcome of the original evaluation up to and including the most recent release (the OpenVPN-NL 2.5 branch). This advisory is intended for ICT decision makers, process owners, security officers and administrators of the product. Based on this advisory, the user organization is able to make a responsible assessment whether or not to use the product in its business process.

Please check on the OpenVPN-NL downloadsite, https://openvpn.fox-it.com (in paragraph "Lifecycle") if you have the latest version of this Deployment Advisory. As soon as an updated version is available, it will be published there.

Requests for more information can be send to the NBV-mailbox NBV@MinBZK.NL

1 Introduction

OpenVPN-NL is a hardened and documented version of the open source OpenVPN. This is a software product for data transport by VPN up to NLNCSA Dep. V. (RESTRICTED).

The evaluation has been performed conform security level 2 of the NLNCSA Evaluation criteria 2020.

This Deployment Advisory describes the protection level of this product and provides information about:
– the evaluated product version;
– the evaluated security features of the product;
– present remaining risks and recommended measures to reduce risks;
– the level of assurance which the evaluation was able to give about the product.

It has been kept as concise as possible. It is advised to read the product information provided by the supplier of the product before reading this advisory. It is strongly recommended to follow the advice that is given in chapter 4 and appendices 1 and 2.

2 Product description

A Virtual Private Network (VPN) is a connection between endpoints that transports network traffic. For data transported by the VPN, the connection between the VPN endpoints appears to be a direct link, when in fact the transported data may travel many steps over multiple external carrier networks. In other words, a VPN creates a new network topology on top of the underlying carrier network(s). Note that VPN tunnels typically connect two machines, but that either endpoint may be the endpoint in multiple connections. In particular, networks may have fully meshed or hub-spoke architectures as well as point to point.

2.1 OpenVPN software

OpenVPN is a software-only VPN implementation that runs as a normal userspace program, as opposed to e.g. IPsec that is typically integrated in the kernel of an operating system. Instead of sending network traffic to a physical network device, OpenVPN attaches to a virtual network device that relays packets to the OpenVPN process – both TAP devices (layer two) and TUN (layer three) devices can be configured. This process performs the set-up of the VPN tunnel and takes care of encryption and decryption of outgoing and incoming packets respectively. Other programs and processes on the platform need not be changed. The setup of the virtual network interface and the adjustment of the operating systems routing tables must be performed with administrative privileges.

An advantage of the implementation as a user space process is the fact that multiple instances of OpenVPN can run simultaneously on the same host. Server and client instances can run side-by-side. Another advantage of a user space application is portability across multiple platforms. All the specific VPN functionality is portable, leaving only the generic low interfacing for each individual operating system.

OpenVPN builds a VPN on top of UDP or TCP. The use of standard higher-level protocols allows OpenVPN tunnels to be easily handled by traditional firewall and Network Address Translation (NAT) systems. The security model of OpenVPN can be divided into three parts:

1. mutual authentication of OpenVPN endpoints based on the TLS/SSL protocol *and on pre-shared keys (administrators must deploy tls-crypt(-v2); see paragraph "*Use tls-crypt(-v2)*");*
2. a secure control channel *(itself protected by pre-shared keys)* to set up and manage VPN tunnels, that is multiplexed with:
3. a secure data channel to transport VPN tunnel payload using a dedicated protocol similar to IPsec's ESP.'

2.1.1 Main differences between vanilla OpenVPN and OpenVPN-NL

The following are the most important differences between vanilla OpenVPN and OpenVPN-NL:

	vanilla OpenVPN	**OpenVPN-NL**
Maintainer	OpenVPN community	Fox-IT
Distribution channel	Various means	https://openvpn.fox-it.com, offline fingerprints available
Certification	None	NLNCSA criteria Level 2, "Dep. VERTROUWELIJK" (Dutch) if deployed in compliance with the Deployment Advisory
Functionality	Full	Many insecure and less secure options stripped, hardened, otherwise unchanged
Cryptographic library	OpenSSL (usually)	mbedTLS

Apart from these differences, many things are the same. The functionality is for the most part identical. Vanilla OpenVPN and OpenVPN-NL are mutually compatible, in the sense that a client of the one can connect to a server of the other (given that a crypto suite is chosen which is available in both products). Configuration files can be freely exchanged (given that the options are available in both products). Both vanilla OpenVPN and OpenVPN-NL and underlying libraries are licensed under GPLv2.

OpenVPN-NL is fully compatible with the OpenVPN protocol, in particular no incompatibilities have been intentionally added.

2.2 System

The security of an OpenVPN-NL deployment depends on several factors:

- the security of the OpenVPN-NL software;
- the configuration of the OpenVPN-NL software;
- the security of the machines/OSes that run the OpenVPN-NL software;
- the security of the key management systems.

When one of these factors fails, the security of the whole deployment falls apart. Therefore, this evaluation will take all these factors into consideration.

2.3 Components

OpenVPN-NL	a software product that is installed on a platform to provide a VPN service.
Configuration	settings for correct operation of OpenVPN and its platform, typically enumerated in a configuration file. Minimal configuration specifies whether an implementation acts as a client or a server.

mbedTLS	A software TLS implementation designed to have a small footprint. The mbedTLS library has been integrated into OpenVPN as part of the hardening, instead of using the much more complex OpenSSL implementation.
pkcs11-helper	plugin software library that enables OpenVPN to use certificates from a smartcard (and performing cryptographic operations on the smartcard) rather than reading the certificates from a regular filesystem.

2.4 Infrastructure

Platform	host with an operating system (Linux and Windows are supported).
PKI	existing Public Key Infrastructure that provides the X.509 certificates needed to establish a connection between an OpenVPN client and server, OpenVPN can be run with an ad hoc PKI (this is not recommended when a suitable PKI at security level two according to the NLNCSA criteria is available).
Key generation station	a system to generate the keys for • tls-crypt-v2 (one server key, plus one key per client), and/or • tls-crypt (one group key), and/or • tls-auth (one group key). For small deployments, customers *may* wish to use the OpenVPN-NL server as the key generation station; as noted in paragraph 4.6, "Use tls-crypt(-v2)", a compromise of the OpenVPN-NL server has an impact on all clients even if the key generation station is separate. Customers that choose to use the OpenVPN-NL server as the key generation station • *may* choose to also implement the ad-hoc PKI on the OpenVPN-NL server, and • *may* choose to generate the Diffie-Hellman parameters on the OpenVPN-NL server. (Note that Diffie-Hellman parameters only need to be generated if non-elliptic-curve Diffie-Hellman is enabled – see "Appendix 1 - OpenVPN-NL options".)
Black IP network	the network that carries the VPN tunnels. This network is deemed "black" from the OpenVPN perspective.
Red IP network	protected network in situations where OpenVPN is used on a gateway to provide a secure connection to a remote network or host.

2.5 Classification

OpenVPN-NL is evaluated with the classification Dep. V. (RESTRICTED) as reference, this corresponds with security level 2 as defined in the NLNCSA criteria. Parts of the system carry the following classifications:

Unclassified	OpenVPN source code, IP network

Unclassified, but sensitive	Configuration, Platform without keys
Departementaal Vertrouwelijk	Platform with keys, red network

The source code for the vanilla OpenVPN is not classified and the patches for the hardened version have been donated to the OpenVPN project. Both should be regarded as known to potential attackers. Local configuration and hardening details should not be made public without an explicit need to do so.

2.6 Deployment scenarios

The evaluation investigates two primary scenarios: peer-to-peer configurations where a client and a server instance of OpenVPN form a single tunnel as peers, and hub spoke configurations where a concentrator acts as server for multiple clients. In the second scenario a red network behind the concentrator is implied. The peer-to-peer scenario includes cases where the peers are actually gateways between red networks, i.e. function as conventional IP crypto devices.

Alternate scenarios, that are not investigated in depth in this report, include deployment of OpenVPN as an add-on product on user machines to allow users to access restricted networks while simultaneously being openly connected to the black carrier network. This allows e.g. surfing the internet while simultaneously being connected to a restricted network. This mode of operation is called 'split tunneling'.

The use of split tunneling on a hostile network brings serious risks as attackers can easily target the host machine. Use of split tunneling on a trusted network (to build a tunnel between two endpoints in a trusted network) does not have these risks.

Split tunneling is not recommended unless it is used to create an exclusive channel over a trusted network.

The PKI infrastructure, that is used for authentication and session set up, is a vital part of the security offered by OpenVPN. Therefore private key material used in this phase needs to be protected. The OpenVPN software supports two modes: using smartcards to store the private key material and perform any operations with that key material – i.e. the private key material never leaves the smartcard – and using a software implementation that uses private key material stored on the filesystem.

Use of smartcards that store the private key material and perform all operations with this key material is recommended.

Use of private key material on the filesystem and/or operations on this key material in software are not recommended, unless the security mechanisms can be shown to offer equivalent security.

3 Level of protection

3.1 Criteria

NLNCSA has evaluated a hardened version of the OpenVPN code base according to the NLNCSA criteria for Dep. V. (RESTRICTED), in conjunction with attackers with a high attack potential. The choice for the high attack potential is based on the fact that OpenVPN is intended to provide communications security over public networks. An additional reason is the fact that the code base is known to attackers due to the fact that OpenVPN is open source and publicly available.

The criteria as used in this evaluation are: "NLNCSA Evaluation Criteria 2020 for securing technology protecting NL classified information", 1 October 2020.

3.2 Evaluated security functions

The hardened version of OpenVPN has been evaluated as a component to provide communications security over public networks. This includes cases where OpenVPN is used as a drop in security measure.

As OpenVPN depends on the host platform for its own security and correct functioning, requirements for the platform have been investigated.

3.3 Absent or non-guaranteed security functions

Although the evaluation provides guidance and recommendations for deployment and requirements for the environment, e.g. regarding routing, firewall settings and assumptions about the PKI architecture, the quality of the implementation of these underlying mechanisms is not part of the evaluation proper. Though these aspects are not part of the evaluation, note that these functions are vital for the overall security of security solutions that incorporate OpenVPN-NL.

3.4 Quantum security

Experts consider the probability small but real that quantum computers will already be powerful enough by 2030 to break current cryptographic standards. The data you send now in encrypted form can be intercepted and decrypted at a later time by a quantum computer. For more information, see NLNCSA's publications "Prepare for the threat of quantum computers" (or the Dutch "Bereid je voor op de dreiging van quantum computers"), both September 2021.

In this OpenVPN-NL deployment advisory, NLNCSA requires that users protect the (quantum-vulnerable) TLS channel by a pre-placed (tls-crypt-v2 or tls-crypt) symmetric key. This protects against quantum attacks, but imperfectly – an attacker who captures the tls-crypt(-v2) key can use a quantum computer to decrypt previously-captured traffic. See paragraph "Lifecycle". This layer of quantum security only exists if a client enables tls-crypt(-v2).

3.5 Disclaimer

The evaluation is not a complete code review. The hardening and documentation process carried out by Fox-IT is intended to minimize the need for such reviews.

The evaluation does not consider an actual deployment and therefore can only provide requirements and guidance on how to securely deploy OpenVPN-NL in generic terms. A user organization needs to perform its own assessment on systems

that build on OpenVPN-NL or incorporate OpenVPN-NL. OpenVPN-NL is a building block, not a total security solution.

4 Guidance on policy and management

4.1 Risks that need to be accepted

OpenVPN-NL is a software product that relies on an operating system and its settings to function correctly. This means that the whole system is susceptible to bugs, configuration errors, incompatibilities between software components and unexpected interactions between hardware and software.
Although open source software means that there are many people identifying bugs and fixing them and various public audits are performed by other parties, it also implicates that attackers can study it in detail.

Paragraph 3.3 'Absent or non-guaranteed security functions' and paragraph 3.5 'Disclaimer' contain a number of – unavoidable – limitations of the OpenVPN-NL and limitations of the evaluation. These limitations lead to residual risks and, before the product is used, informed choices will have to be made about this. The most important choice lies in balancing of the hardening and configuration of the host platform and functional requirements.

4.2 Distribution

An evaluation by the NLNCSA covers many aspects, including cryptographic aspects, source code inspection and supply chain control. The OpenVPN releases as found on the openvpn.net website or elsewhere on the net or the installation disk of various operating systems, though possibly of great protection value, cannot get an approval of the NLNCSA. The two most important reasons are:

1. The product allows many insecure configurations, including, but not limited to, turning off encryption, or the use of outdated cryptographic functions at security critical places.
2. Trust in the supply chain of the software is not guaranteed. The Dutch government cannot verify whether all the versions and releases 'out in the wild' are legitimate (i.e. secure and uncompromised) versions of OpenVPN.

Please note that this list of reasons is not exhaustive.

To address these issues, NLNCSA has commissioned Fox-IT to create a special Dutch version of OpenVPN: OpenVPN-NL. Fox-IT has stripped and hardened the product, and has set up a controlled distribution channel on https://openvpn.fox-it.com. Fox-IT has acted as the maintainer and distributor of OpenVPN-NL under direction of the NLNCSA, and will do so for the foreseeable future.

OpenVPN-NL meets the evaluation criteria of the NLNCSA for handling classified information up to the level of Dep. V. (RESTRICTED). Safe use of OpenVPN-NL requires compliance with the conditions set in this deployment advisory, published by the NLNCSA.

NLNCSA has evaluated the protection offered by the product against breaches of confidentiality and integrity. The evaluation has not included availability (e.g. robustness) of the VPN tunnel provided by OpenVPN-NL.

4.2.1 Practical actions

System administrators who wish to use OpenVPN-NL are advised to:

- Download the latest OpenVPN-NL for their platform from the website https://openvpn.fox-it.com;

- Verify the fingerprint of the downloaded package on the NLNCSA website: https://www.aivd.nl/onderwerpen/informatiebeveiliging/beveiligingsproducten/inzetadviezen;
- Subscribe to the OpenVPN-NL mailinglist;
- Create a system which is in compliance with the requirements of this Deployment Advisory;
- Install OpenVPN-NL on this system in compliance with this Deployment Advisory.

4.2.2 New releases

The following events can trigger new releases:

- a new release of one of the packages on which OpenVPN-NL is based;
- a new release of one of the platforms for which OpenVPN-NL is packaged.

Releases which address security issues will be released as quickly as possible, other releases will be dealt with at a more planned pace.
In line with the mainstream OpenVPN policy, new releases of OpenVPN-NL will if anywhere possible, be backwards compatible in the sense that two different OpenVPN(-NL) releases can connect to one another (they do not break the protocol), and that new OpenVPN(-NL) releases can read the configuration files of older versions of OpenVPN(-NL). For one example, see paragraph "Key sizes: recommendations and requirements for interoperability".

4.2.2.1 Announcements

When a new release of OpenVPN-NL is available, this will always be announced by means of the OpenVPN-NL mailing list. This is a low-volume read-only mailing list for this purpose only. Announcements include a statement on whether the new release addresses security issues.

4.2.3 Evaluated version

On the distribution website https://openvpn.fox-it.com/, users can find the compiled installable packages.

Users interested in digging deeper can also find the source code (which is not itself supported), and can find the versions of vanilla OpenVPN-NL and libraries used to build the OpenVPN-NL releases.

4.2.4 Lifecycle

A specific release of OpenVPN-NL for a platform is always in one of the following three states:

- *Current*: this release is up-to-date. This one is the recommended release for that platform.
- *Deprecated*: this is an old version of OpenVPN-NL for that platform, which is not up-to-date, but has no security issues. If a current version exists for the platform, that version is recommended for general reasons. There are no short-term security-related reasons to advise against using a deprecated version.
- *Untrusted/insecure*: this is an old release of OpenVPN-NL, for which reasons exist to very strongly advise against using it.
  There are two typical cases:
  - The release which has become that much different from the current version of OpenVPN-NL, that it is no longer practical to determine

whether a vulnerability found in a current (or deprecated) release applies to the old release.
    - The version has known vulnerabilities or is for other reasons sufficiently suspect to very strongly advise against using it.

The state of a release always moves down (from "Current" via "Deprecated" to "Untrusted"), and may skip the deprecated state. For example, a current release might directly become untrusted/insecure, skipping the deprecated state. A release in the deprecated state may (quickly) advance into the untrusted/insecure state.

On the OpenVPN-NL website, current and deprecated releases will be available for download, but untrusted/insecure versions and their corresponding source code will never be available for download. Untrusted/insecure versions or corresponding source code will also not be available via other channels from NLNCSA or Fox-IT.

Under all circumstances, NLNCSA advises to use a current OpenVPN-NL release. When system administrators depend on a specific release of OpenVPN, they should be aware that it may become untrusted/insecure at some time in the future and, therefore, no longer available for download. While work-arounds might exist which mitigate the vulnerabilities in a specific context to some extent, hanging on to an old code base is discouraged.

When an OpenVPN-NL release moves into the untrusted/insecure state, this will be announced on the OpenVPN-NL mailing list. This announcement will typically be complemented with the announcement of a new release.

4.2.5 Key sizes: recommendations and requirements for interoperability

In the deployment advisory v1.4, the recommended length of the RSA and EDH moduli was 2048 bits. Starting with the deployment advisory for OpenVPN 2.5.x, NLNCSA requires users to use 3072 bits for both RSA and EDH moduli. NLNCSA continues to recommend ECDHE with a 256-bit curve.

NLNCSA recognizes that organizations need a transition period, and currently thinks this transition would best be handled as follows, to ensure that upgrading the OpenVPN-NL software by a minor release (2.4.x to 2.5.x, 2.5.x to 2.6.x, etc.) will either result in a working network or cause the local client to reject the configuration (but will not cause the peer to reject the configuration.)

- In OpenVPN-NL 2.5.x NLNCSA requires in the deployment advisory, but does not technically enforce, that users generate suitable (i.e. RSA-3072 / DH-3072 or better) key material. OpenVPN-NL 2.5.x. comes with suitable documentation and supporting software to e.g. ensure that generating RSA private keys using any provided scripts results in RSA-3072 bit keys (unless explicitly overridden by the user), and OpenVPN-NL logs a warning when using shorter private keys.

- OpenVPN-NL 2.6.x refuses to start when the private key material is too short (i.e. requires RSA keys to be at least 3072-bit); and (as in 2.5.x,) successfully negotiates a connection when the peer uses RSA and/or EDH with n-bit keys, for 2048 <= n.

- OpenVPN-NL 2.7.x (as in 2.6.x,) refuses to start when the private key material is too short (i.e. requires RSA keys to be at least 3072-bit); and refuses to negotiate a connection when the peer uses RSA and/or EDH with n-bit keys, for n < 3072.

4.3 Management

Management falls into three categories: management of OpenVPN itself, management of the platform that runs OpenVPN, and key management. The evaluation identifies the following requirements based on the evaluation results:

Cryptosuite

The table below mentions the cipher suites that are supported in OpenVPN-NL 2.5.x for both the data channel and the control channel. All supported cryptosuites are appropriate for OpenVPN-NL.

Channel	Cipher	Remark
Data Channel	AES-256-GCM	Default
Data Channel	AES-256-CBC	Deprecated in OpenVPN-NL 2.5.x, and removed in OpenVPN-NL 2.6.x
Control Channel	TLS-ECDHE-RSA-WITH-AES-256-GCM-SHA384	Implement by using a 256 or 384-bit elliptic curve and 3072-bit moduli for both EDH and RSA
Control Channel	TLS-DHE-RSA-WITH-AES-256-GCM-SHA384	Implement by using a 256 or 384-bit elliptic curve and 3072-bit moduli for both EDH and RSA
Control Channel	TLS-DHE-RSA-WITH-AES-256-CBC-SHA256	Implement by using a 256 or 384-bit elliptic curve and 3072-bit moduli for both EDH and RSA

Public Key Infrastructure

OpenVPN must be configured to use an existing or ad-hoc PKI implementation at security level two of the NLNCSA Criteria. This corresponds to "Departementaal Vertrouwelijk". Client and servers must only accept certificates corresponding to appropriate counterparties; for an ad-hoc (i.e. dedicated) PKI, it may suffice to simply accept all server respectively all client certificates signed by the ad-hoc Certificate Authority (CA). See "Appendix 1 - OpenVPN-NL options".

Platform

The platform must be configured to monitor and log unexpected traffic or attempts to bypass routing, firewall rules or system policies.
The platform must be hardened according to industry and government standards.
The platform must be patched with all relevant security patches provided by the manufacturer.

Configuration

tls-crypt(-v2) must be enabled, see paragraph "use tls-crypt(-v2)".
OpenVPN must be configured to use the least amount of privileges needed to run. See, again, "Appendix 1 - OpenVPN-NL options".

Routing

The platform's routing tables must preclude any route that bypasses OpenVPN.

Firewall
The platform's firewall rules must preclude any traffic other than the traffic to and from OpenVPN or traffic required to let OpenVPN function.
The platform must only accept traffic from known peers in the VPN the platform belongs to.

Reference for these requirements is security level two according to the NLNCSA criteria. When any aspect of these requirements cannot be met, or not be met in full, in specific OpenVPN deployments, the responsible security officer should be consulted, who can relay specific questions to the NLNCSA.

4.4 Instructions for users

End-users of OpenVPN-NL generally need no additional instruction for the product itself as the operation of OpenVPN-NL is mostly transparent to end-users. End-users do need to be instructed on basic security aspects such as how to handle credentials – e.g. passwords, smartcards, certificates – needed to access the host that runs OpenVPN-NL, and how to start the VPN tunnel where needed. Instructing users on how to recognize correct behavior and where to report deviations or incidents, is recommended.

4.5 Use – general guidance

OpenVPN allows many settings to be configured. Some of the options have been eliminated as part of the hardening patches. Using configuration files rather than command line options is recommended for deployment outside of test environments. These configuration files should be made read only and stored such that a minimum subset of processes can access these files.

Log files should likewise be protected from unnecessary access through restrictions on access rights for these files. Log levels should be set no higher than necessary to support operational or other security needs (e.g. determining which users may have taken a particular malicious action).

The use of UDP as the carrier protocol for OpenVPN is recommended, unless only TCP can be used.

OpenVPN-NL does not support compression.

The use of TUN or TAP devices for the virtual network device is at the user's discretion. This choice needs to be made for all VPN endpoints as TUN and TAP are mutually incompatible (they operate at different levels in the OSI stack).

OpenVPN now supports VLANs between server and clients, however NLNCSA does not recommend using VLANs for domain separation.

Systems of customers, as with all IT security solutions at this security level, shall comply with (the requirements on TEMPEST protection in) SDIP-27, Level C; or customers shall accept the corresponding risk.

Customers must distribute all key material securely, with measures appropriate to their risk.

4.6 Use tls-crypt(-v2)

Use of the tls-crypt(-v2) function is required for NLNCSA approval. Unless you specifically need compatibility with tls-crypt (i.e. with OpenVPN-NL 2.4.x), use tls-crypt-v2 to ensure that, when a client-specific key is compromised, this will only affect one client instead of all clients in the case in which tls-crypt (v1) is used.

A roll-over mechanism and subsequent policy for compromised keys need to be in place. This policy needs to entail the revocation of compromised keys, distribution of new keys after compromise and removal of compromised keys. For configurations supporting some possible policies, see "Appendix 1 - OpenVPN-NL options", "Options required for OpenVPN-NL".

NLNCSA recommends updating the tls-crypt-v2 keys *frequently* in order to minimize the impact when a key is lost; once the key is lost, a (future) quantum adversary can decrypt recorded traffic. The table below specifies the impact corresponding to the lost key.

Key lost	Impact on
tls-crypt-v2 client-specific key	specific client
tls-crypt-v2 server key	all clients
tls-crypt group key	all clients

As an illustration, to ensure that losing tls-crypt-v2 keys threatens at most one year of traffic,

- An organization relying on OpenVPN-NL to protect its own infrastructure may choose to simply replace all tls-crypt-v2 keys and TLS certificates in a single yearly upgrade; whereas
- An organization relying on OpenVPN-NL for remote workers may choose to:
    - Once a year, set up a OpenVPN-NL server daemon with a new tls-crypt-v2 server keys (this requires a unique port, but will function when using the same IP address and server hardware); and
    - For each OpenVPN client, at some convenient time for that client not more than one year after the latest refresh, generate (a new TLS certificate,) a tls-crypt-v2 client-specific key (for the latest tls-crypt-v2 server key), and an OpenVPN-NL configuration pointing at the latest OpenVPN server; and
    - Decommission the old client configuration before it has been in use for a year; decommission the old server once all clients have been drained (i.e. after at most a two year-period, in which each individual client has used the key for at most one year).

It is up to the customer of OpenVPN-NL to determine this frequency (dependent on their use-case). Government or vital infrastructure customers may contact the NLNCSA for advice.

Customers who are unable to replace tls-crypt-v2 keys frequently, should use tls-crypt-v2 either way. In addition to the protection against quantum computers offered by tls-crypt(-v2), using any of tls-crypt-v2, tls-crypt or tls-auth also protects against attacks on the TLS protocol and/or on the OpenVPN protocol negotiation. NLNCSA and Fox-IT handling of security updates assumes that customers have implemented the requirements in this deployment advisory. **Do not disable tls-crypt-v2**.

4.7 Ensuring OpenVPN-NL has enough entropy

On Linux kernels older than version 5.6 (released 29 March 2020), OpenVPN-NL servers with many clients have been observed to stall (during connection establishment) until additional entropy becomes available.

The NLNCSA does not consider such stalling a *security* risk (except possibly to availability), but most system administrators will wish to ensure the system does not stall. System administrators may wish to consult the guidance in appendix 2.

4.8 Incidents

Incidents can occur in the context of communication security and system security.

Communication security incidents imply the loss of security functionality of a single connection, this would be the case where an attacker gains access to key material of some kind or the encryption mechanisms fail. See also paragraph 4.6 'Use tls-crypt(-v2)'

The underlying PKI infrastructure is assumed to provide procedures for revocation of compromised smartcards.

System security incidents where an attacker gains access to the system that hosts OpenVPN are more serious. Unauthorized access may compromise the software or the operating system, which leads to complete breakdown of security. The requirements that pertain to monitoring of the system, its users, and the dataflow, help detect possible incidents early on.

5 Conclusions

OpenVPN-NL is suitable as a building block for securing communication and access to systems to NLNCSA Evaluation criteria 2020 up to Dep. V. (RESTRICTED), security level 2, given that the requirements detailed in this report are correctly implemented, notably that:

- the underlying platform is secure and configured such that OpenVPN is in charge of all incoming and outgoing traffic, and
- a PKI conforming to security level two of the NLNCSA criteria is used for authentication, and
- tls-crypt(-v2) is used (see paragraph 4.6 ‘Use tls-crypt(-v2)’.)

Please note that while correct implementation of the requirements detailed in this report provides sufficient assurance, NLNCSA cannot guarantee the platform and its configuration without verification.

These conclusions are limited to the versions that are mentioned on the OpenVPN-NL download site, section "Lifecycle" (https://openvpn.fox-it.com/lifecycle.html). All deviations from this advice should be acknowledged and documented by the organisation that deploys OpenVPN-NL, to enable proper risk management.

This evaluation addresses the cases of *peer-to-peer* and *hub-spoke* networks. The conclusions for both scenarios are identical, with the caveat that in a hub-spoke network, the hub or concentrator may provide a high profile target for attacks. Hub-spoke networks sometimes consist of loosely managed platforms, i.e. the nodes on the spokes are likely not under direct control of the system administrators responsible for the hub and the VPN.

Appendix 1 - OpenVPN-NL options

This appendix is meant as checklist for the network operator to create a proper configuration for a sufficient secure VPN connection. Only the security related options are listed here. The next paragraphs contain options that must be specified and options that should at least be considered to be included for further hardening the VPN. More information about the various options can be found in the public OpenVPN documentation (see https://openvpn.net).

Options required for OpenVPN-NL
The configuration of a secure OpenVPN-NL connection must at least specify all of the following options:

- --tls-crypt-v2 *keyfile* or --tls-crypt *keyfile* (see par. 4.6 'Use tls-crypt(-v2)'.)
- --cert *file*
- --key *file*

The configuration must additionally specify options that ensure both client and server are connecting to appropriate counterparties. Possible configurations include:

- (for a PKI in which all clients are trusted to connect to the OpenVPN server(s),)
    - (on the server,) --ca *file* --remote-cert-tls client, respectively
    - (on the client,) --ca *file* --remote-cert-tls server --verify-x509-name *name type*.

Government or vital infrastructure clients may contact NLNCSA for (further) assistance.

The configuration must additionally specify options required to implement the administrator's/organization's policy for the revocation of compromised keys. Some possible policies, and supporting configurations, are:

- no options at all.

  This configuration supports the following sample policy:
    - In case any OpenVPN-NL system, or any other system holding a certificate from this PKI, is compromised, regenerate the entire PKI including the Certificate Authority, regenerate all tls-crypt(-v2) keys, and use some mechanism external to OpenVPN-NL to overwrite the old keys with the new keys on all systems. (That is, revoke all certificates by replacing the Certificate Authority, and revoke all tls-crypt(-v2) keys by replacing the tls-crypt(-v2) keys.)
- (for a PKI in which all clients are trusted to connect to the OpenVPN server(s), on the server only,)
  --crl-verify *file_or_directory* [dir]

  This configuration supports the following sample policy:
    - In case any server or key generation station is compromised, (as above) regenerate and overwrite all keys;
    - in case an OpenVPN-NL client is compromised,
        - revoke the client certificate by placing the client certificate on the Certificate Revocation List (CRL) file on the server, and
        - (if/when the client should have access to the protected network again,) regenerate the client certificate and tls-

crypt-v2 client key, and use some mechanism external to OpenVPN-NL to place the new client certificate and tls-crypt-v2 client key on the client (no changes are required to the server).
        - This configuration does not support revocation of (potentially-)compromised tls-crypt-v2 client keys. Instead, other parts of the policy reduce the risk to a level that the relevant authority can accept (and **the relevant authority must explicitly accept this deviation from the requirement to revoke compromised keys in par. 4.6, "Use tls-crypt(-v2)"**), e.g. by
            - frequent rotation of the tls-crypt-v2 server key (which revokes all tls-crypt-v2 client keys), and/or by
            - using tls-crypt-v2, which (unlike the original tls-crypt) has per-client keys.

          See par. 4.6, "Use tls-crypt(-v2)"; in particular, note that "[C]ustomers who are unable to replace tls-crypt-v2 keys frequently should use tls-crypt-v2 either way".
- (for a PKI in which all clients are trusted to connect to the OpenVPN server(s), on the server only,)
  --crl-verify *file_or_directory* [dir] --tls-crypt-v2-verify *program_to_be_provided_by_customer*

  This configuration supports the following sample policy:
    - (as above, except) in case an OpenVPN-NL client is compromised,
        - (as above, except) *do* revoke the tls-crypt-v2 client key, by causing *program_to_be_provided_by_customer* to reject the tls-crypt-v2 client key (based on its metadata). Note that this requires
            - generating the tls-crypt-v2 client keys with suitable metadata (e.g. the client certificate serial number plus CA fingerprint, as suggested in the OpenVPN-NL documentation), and a
            - *program_to_be_provided_by_customer* to consult some list of tls-crypt-v2 client key revocations (e.g. the CRL file).

          The current OpenVPN-NL release does not include a (sample) tool to implement this lookup ("*program_to_be_provided_by_customer*").

          OpenVPN-NL --genkey tls-crypt-v2-client *keyfile [metadata]* can be used to generate a tls-crypt-v2 client key with any administrator-specified *metadata*, but the current OpenVPN-NL release does not include a (sample) tool that assists in choosing and/or recording *metadata* suitable for any particular revocation policy.

Again, government or vital infrastructure clients may contact NLNCSA for (further) assistance.

Recommended options
For further hardening OpenVPN at least following options should be considered to be part of the configuration:

- NLNCSA recommends protecting the system running OpenVPN-NL by dropping privileges:

- --user *dedicated_user*; and
- --group *dedicated_group*

  where *dedicated_user* (resp. *dedicated_group*) represents a configured unprivileged user (esp. group). The option changes the user (or group) of the OpenVPN process after initialization. If applied, --persist-key and --persist-tun should be considered as well.

- NLNCSA recommends further protecting the system running OpenVPN-NL by restricting (file-)system access via --chroot *dir* and/or even --setcon *SELinux_context*. As above, consider --persist-key and --persist-tun.
- Do not enable the remote management interface (--management). While most security issues with this interface have been tackled, normal installations do not need this management interface to be active. Not turning it on (which is the default) is strongly recommended for production environments.
- NLNCSA recommends that administrators not enable more functionality than required for their network.
  - *Consider* not configuring the client to allow the server to push settings (i.e. omit --pull from the client configuration; this usually means specifying --tls-client instead of --client, since --client is equivalent to --pull --tls-client.)

    OpenVPN has not had any vulnerabilities in its handling of --pull for many years, but --pull *does* open some attack surface.
  - *Consider* setting --script-security 0; external programs that are never executed can't cause any issues.
- On Windows hosts (likely clients), consider --block-outside-dns to ensure that the system does not leak which hosts are accessed by sending DNS queries outside the VPN tunnel.
- NLNCSA recommends that administrators enable the option --use-prediction-resistance, which causes the internal RNG to reseed in each call for random. Using this option ensures that an attacker compromising the OpenVPN process does not obtain key material related to previous (or future) connections.

  When using this option with a Linux kernel older than version 5.6, consider running a daemon that adds entropy to the kernel pool; see "Appendix 2 - hints for system administrators".

Options related to networking:

- The NLNCSA does not consider the use of VLANs as a strong security measure in general and therefore this functionality (--vlan-tagging, --vlan-accept, --vlan-pvid) was not evaluated. An organization that may want to use VLANs as a means of internal network separation, should be aware of the limited security value.
- To help clients (using the default UDP protocol) to reconnect quickly when the OpenVPN server is rebooted, consider --explicit-exit-notify *[n]*

Options with respect to monitoring (can be used to) enhance visibility on the operational status of the OpenVPN-NL process or client connections. *Consider* using these.

- --log *file*, --log-append *file*, --syslog or --daemon
- --verb *number* (but see par. 4.5 'use – general guidance')
- --status *[n]* [--status-version *n*]
- --client-connect *script*

- --ipchange *script*

For compatibility, you *may* wish to consider

- --dh *file*: *only* if you need, for compatibility reasons, to support the EDH TLS ciphersuites in addition to the ECDH TLS ciphersuites. OpenVPN-NL has supported the TLS ECDH ciphersuites since at least 2.3.x. *If* you do use this option, generate a *file* suitable for 3072-bit EDH, e.g. by `openssl dhparam -out dh3072.pem 3072`. (See par. 4.2.5, "Key sizes: recommendations and requirements for interoperability".)

Appendix 2 - hints for system administrators

As noted in Paragraph 4.7 'Ensuring OpenVPN-NL has enough entropy', on Linux kernels older than version 5.6 (released 29 March 2020), OpenVPN-NL servers with many clients have been observed to stall (during connection establishment) until additional entropy becomes available.

The NLNCSA does not consider such stalling a *security* risk (except possibly to availability). Therefore, NLNCSA places no related requirements on the deployment of OpenVPN-NL.

System administrators, however, *may* find the remarks in this appendix useful.

OpenVPN-NL stalls when no entropy is available from the /dev/random kernel device. If such stalls are observed, system administrators *may* wish to ensure that sufficient entropy is available, e.g. by

- deploying on Linux kernel version 5.6 or later (which will generate additional entropy as needed), *or* by
- adding a hardware random number generator.

Appendix 2.1 Linux kernel versions

As of late 2022, Linux kernel versions 5.6 or later appear to be available for all Linux distributions for which OpenVPN-NL releases are built, i.e. in

- Debian 11 "Bullseye", and also in Debian 10 "Buster" as a so-called "backport"; and in
- Ubuntu 20.04 and later, and also in Ubuntu 18.04 via the "hardware enablement" kernel; and in
- Red Hat Enterprise Linux 9; and in
- SUSE Linux Enterprise 15.

Administrators are directed to their Linux distributions' documentation and/or support for further details and for an authoritative list of supported kernel versions.

Appendix 2.2 Additional background information

The (extremely!) interested reader may wish to consult the Linux Weekly News article "Removing the Linux /dev/random blocking pool" at https://lwn.net/Articles/808575 ("reads from /dev/random (…) will not block and will return the requested amount of random data"), or even the accepted patch "random: make /dev/random be almost like /dev/urandom" (Linux kernel commit 30c08efec), which generates additional entropy via "Merge branch 'entropy'" (Linux kernel commit 3f2dc2798b).