SSA-093430: Multiple Vulnerabilities in SIMATIC RTLS Locating Manager before V3.0

Publication Date: 2024-05-14
Last Update: 2024-06-11
Current Version: V1.1
CVSS v3.1 Base Score: 10.0
CVSS v4.0 Base Score: 10.0

SUMMARY

Siemens has released a new version for SIMATIC RTLS Locating Manager and recommends to update to the latest version.

AFFECTED PRODUCTS AND SOLUTION

Affected Product and Versions	Remediation
SIMATIC RTLS Locating Manager (6GT2780-0DA00): All versions < V3.0.1.1 affected by all CVEs	Update to V3.0.1.1 or later version The update is available from Siemens Online Software Delivery (OSD). See further recommendations from section Workarounds and Mitigations
SIMATIC RTLS Locating Manager (6GT2780-0DA10): All versions < V3.0.1.1 affected by all CVEs	Update to V3.0.1.1 or later version The update is available from Siemens Online Software Delivery (OSD). See further recommendations from section Workarounds and Mitigations
SIMATIC RTLS Locating Manager (6GT2780-0DA20): All versions < V3.0.1.1 affected by all CVEs	Update to V3.0.1.1 or later version The update is available from Siemens Online Software Delivery (OSD). See further recommendations from section Workarounds and Mitigations
SIMATIC RTLS Locating Manager (6GT2780-0DA30): All versions < V3.0.1.1 affected by all CVEs	Update to V3.0.1.1 or later version The update is available from Siemens Online Software Delivery (OSD). See further recommendations from section Workarounds and Mitigations
SIMATIC RTLS Locating Manager (6GT2780-1EA10): All versions < V3.0.1.1 affected by all CVEs	Update to V3.0.1.1 or later version The update is available from Siemens Online Software Delivery (OSD). See further recommendations from section Workarounds and Mitigations

SIMATIC RTLS Locating Manager (6GT2780-1EA20): All versions < V3.0.1.1 affected by all CVEs	Update to V3.0.1.1 or later version The update is available from Siemens Online Software Delivery (OSD). See further recommendations from section Workarounds and Mitigations
SIMATIC RTLS Locating Manager (6GT2780-1EA30): All versions < V3.0.1.1 affected by all CVEs	Update to V3.0.1.1 or later version The update is available from Siemens Online Software Delivery (OSD). See further recommendations from section Workarounds and Mitigations

WORKAROUNDS AND MITIGATIONS

Siemens has identified the following specific workarounds and mitigations that customers can apply to reduce the risk:

- CVE-2024-30207: Protect all communication between RTLS Clients and the Server using a secure channel, e.g. an appropriate VPN solution. Ensure that the configured Server ports are exclusively reachable via the VPN as described in the installation manual
- Install required RTLS Locating Manager components on a single host computer where possible and ensure only trusted persons have access to the system
- Secure the Windows Server, where the RTLS Locating Manager is installed on, with a firewall and make sure no ports are accessible from untrusted networks
- Apply security hardening of the Windows Server, where the RTLS Locating Manager is installed on, in accordance with your corporate security policies or up-to-date hardening guidelines

Product-specific remediations or mitigations can be found in the section Affected Products and Solution. Please follow the General Security Recommendations.

GENERAL SECURITY RECOMMENDATIONS

As a general security measure, Siemens strongly recommends to protect network access to devices with appropriate mechanisms. In order to operate the devices in a protected IT environment, Siemens recommends to configure the environment according to Siemens' operational guidelines for Industrial Security (Download: https://www.siemens.com/cert/operational-guidelines-industrial-security), and to follow the recommendations in the product manuals. Additional information on Industrial Security by Siemens can be found at: https://www.siemens.com/industrialsecurity

PRODUCT DESCRIPTION

SIMATIC RTLS Locating Manager is used for the configuration, operation, and maintenance of a SIMATIC RTLS installation, which is real-time wireless locating system for flexible and cost-effective locating solutions.

VULNERABILITY DESCRIPTION

This chapter describes all vulnerabilities (CVE-IDs) addressed in this security advisory. Wherever applicable, it also documents the product-specific impact of the individual vulnerabilities.

Vulnerability CVE-2023-4807

Issue summary: The POLY1305 MAC (message authentication code) implementation contains a bug that might corrupt the internal state of applications on the Windows 64 platform when running on newer X86_64 processors supporting the AVX512-IFMA instructions. Impact summary: If in an application that uses the OpenSSL library an attacker can influence whether the POLY1305 MAC algorithm is used, the application state might be corrupted with various application dependent consequences. The POLY1305 MAC (message authentication code) implementation in OpenSSL does not save the contents of non-volatile XMM registers on Windows 64 platform when calculating the MAC of data larger than 64 bytes. Before returning to the caller all the XMM registers are set to zero rather than restoring their previous content. The vulnerable code is used only on newer x86_64 processors supporting the AVX512-IFMA instructions. The consequences of this kind of internal application state corruption can be various - from no consequences, if the calling application does not depend on the contents of non-volatile XMM registers at all, to the worst consequences, where the attacker could get complete control of the application process. However given the contents of the registers are just zeroized so the attacker cannot put arbitrary values inside, the most likely consequence, if any, would be an incorrect result of some application dependent calculations or a crash leading to a denial of service. The POLY1305 MAC algorithm is most frequently used as part of the CHACHA20-POLY1305 AEAD (authenticated encryption with associated data) algorithm. The most common usage of this AEAD cipher is with TLS protocol versions 1.2 and 1.3 and a malicious client can influence whether this AEAD cipher is used by the server. This implies that server applications using OpenSSL can be potentially impacted. However we are currently not aware of any concrete application that would be affected by this issue therefore we consider this a Low severity security issue. As a workaround the AVX512-IFMA instructions support can be disabled at runtime by setting the environment variable OPENSSL_ia32cap: OPENSSL_ia32cap=: 0x200000 The FIPS provider is not affected by this issue.

CVSS v3.1 Base Score	7.8
CVSS Vector	CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H/E:P/RL:O/RC:C
CWE	CWE-20: Improper Input Validation

Vulnerability CVE-2023-5363

Issue summary: A bug has been identified in the processing of key and initialisation vector (IV) lengths. This can lead to potential truncation or overruns during the initialisation of some symmetric ciphers.

Impact summary: A truncation in the IV can result in non-uniqueness, which could result in loss of confidentiality for some cipher modes.

When calling EVP_EncryptInit_ex2(), EVP_DecryptInit_ex2() or EVP_CipherInit_ex2() the provided OSSL_PARAM array is processed after the key and IV have been established. Any alterations to the key length, via the “keylen” parameter or the IV length, via the “ivlen” parameter, within the OSSL_PARAM array will not take effect as intended, potentially causing truncation or overreading of these values. The following ciphers and cipher modes are impacted: RC2, RC4, RC5, CCM, GCM and OCB.

For the CCM, GCM and OCB cipher modes, truncation of the IV can result in loss of confidentiality. For example, when following NIST’s SP 800-38D section 8.2.1 guidance for constructing a deterministic IV for AES in GCM mode, truncation of the counter portion could lead to IV reuse.

Both truncations and overruns of the key and overruns of the IV will produce incorrect results and could, in some cases, trigger a memory exception. However, these issues are not currently assessed as security critical.

Changing the key and/or IV lengths is not considered to be a common operation and the vulnerable API was recently introduced. Furthermore it is likely that application developers will have spotted this problem during testing since decryption would fail unless both peers in the communication were similarly vulnerable. For these reasons we expect the probability of an application being vulnerable to this to be quite low. However if an application is vulnerable then this issue is considered very serious. For these reasons we have assessed this issue as Moderate severity overall.

The OpenSSL SSL/TLS implementation is not affected by this issue.

The OpenSSL 3.0 and 3.1 FIPS providers are not affected by this because the issue lies outside of the FIPS provider boundary.

OpenSSL 3.1 and 3.0 are vulnerable to this issue.

CVSS v3.1 Base Score	7.5
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N/E:P/RL:O/RC:C
CWE	CWE-20: Improper Input Validation

Vulnerability CVE-2023-5678

Issue summary: Generating excessively long X9.42 DH keys or checking excessively long X9.42 DH keys or parameters may be very slow. Impact summary: Applications that use the functions DH_generate_key() to generate an X9.42 DH key may experience long delays. Likewise, applications that use DH_check_pub_key(), DH_check_pub_key_ex() or EVP_PKEY_public_check() to check an X9.42 DH key or X9.42 DH parameters may experience long delays. Where the key or parameters that are being checked have been obtained from an untrusted source this may lead to a Denial of Service. While DH_check() performs all the necessary checks (as of CVE-2023-3817), DH_check_pub_key() doesn’t make any of these checks, and is therefore vulnerable for excessively large P and Q parameters. Likewise, while DH_generate_key() performs a check for an excessively large P, it doesn’t check for an excessively large Q. An application that calls DH_generate_key() or DH_check_pub_key() and supplies a key or parameters obtained from an untrusted source could be vulnerable to a Denial of Service attack. DH_generate_key() and DH_check_pub_key() are also called by a number of other OpenSSL functions. An application calling any of those other functions may similarly be affected. The other functions affected by this are DH_check_pub_key_ex(), EVP_PKEY_public_check(), and EVP_PKEY_generate(). Also vulnerable are the OpenSSL pkey command line application when using the “-pubcheck” option, as well as the OpenSSL genpkey command line application. The OpenSSL SSL/TLS implementation is not affected by this issue. The OpenSSL 3.0 and 3.1 FIPS providers are not affected by this issue.

CVSS v3.1 Base Score	5.3
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L/E:P/RL:O/RC:C
CWE	CWE-754: Improper Check for Unusual or Exceptional Conditions

Vulnerability CVE-2023-29409

Extremely large RSA keys in certificate chains can cause a client/server to expend significant CPU time verifying signatures. With fix, the size of RSA keys transmitted during handshakes is restricted to <= 8192 bits. Based on a survey of publicly trusted RSA keys, there are currently only three certificates in circulation with keys larger than this, and all three appear to be test certificates that are not actively deployed. It is possible there are larger keys in use in private PKIs, but we target the web PKI, so causing breakage here in the interests of increasing the default safety of users of crypto/tls seems reasonable.

CVSS v3.1 Base Score	5.3
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L/E:P/RL:O/RC:C
CWE	CWE-400: Uncontrolled Resource Consumption

Vulnerability CVE-2023-33953

PC contains a vulnerability that allows hpack table accounting errors could lead to unwanted disconnects between clients and servers in exceptional cases. Three vectors were found that allow the following DOS attacks: - Unbounded memory buffering in the HPACK parser - Unbounded CPU consumption in the HPACK parser The unbounded CPU consumption is down to a copy that occurred per-input-block in the parser, and because that could be unbounded due to the memory copy bug we end up with a parsing loop, with n selected by the client. The unbounded memory buffering bugs: - The header size limit check was behind the string reading code, so we needed to first buffer up to a 4 gigabyte string before rejecting it as longer than 8 or 16kb. - HPACK varints have an encoding quirk whereby an infinite number of 0's can be added at the start of an integer. gRPC's hpack parser needed to read all of them before concluding a parse. - gRPC's metadata overflow check was performed per frame, so that the following sequence of frames could cause infinite buffering: HEADERS: containing a: 1 CONTINUATION: containing a: 2 CONTINUATION: containing a: 3 etc. . . - Unbounded memory buffering in the HPACK parser - Unbounded CPU consumption in the HPACK parser

The unbounded CPU consumption is down to a copy that occurred per-input-block in the parser, and because that could be unbounded due to the memory copy bug we end up with an parsing loop, with n selected by the client.

The unbounded memory buffering bugs:

- The header size limit check was behind the string reading code, so we needed to first buffer up to a 4 gigabyte string before rejecting it as longer than 8 or 16kb.
- HPACK varints have an encoding quirk whereby an infinite number of 0's can be added at the start of an integer. gRPC's hpack parser needed to read all of them before concluding a parse.
- gRPC's metadata overflow check was performed per frame, so that the following sequence of frames could cause infinite buffering: HEADERS: containing a: 1 CONTINUATION: containing a: 2 CONTINUATION: containing a: 3 etc. . .

CVSS v3.1 Base Score	7.5
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H/E:P/RL:O/RC:C
CWE	CWE-834: Excessive Iteration

Vulnerability CVE-2023-38039

When curl retrieves an HTTP response, it stores the incoming headers so that they can be accessed later via the libcurl headers API.

However, curl did not have a limit in how many or how large headers it would accept in a response, allowing a malicious server to stream an endless series of headers and eventually cause curl to run out of heap memory.

CVSS v3.1 Base Score	7.5
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H/E:P/RL:O/RC:C
CWE	CWE-770: Allocation of Resources Without Limits or Throttling

Vulnerability CVE-2023-38545

This flaw makes curl overflow a heap based buffer in the SOCKS5 proxy handshake.

When curl is asked to pass along the hostname to the SOCKS5 proxy to allow that to resolve the address instead of it getting done by curl itself, the maximum length that hostname can be is 255 bytes.

If the hostname is detected to be longer than 255 bytes, curl switches to local name resolving and instead passes on the resolved address only to the proxy. Due to a bug, the local variable that means "let the host resolve the name" could get the wrong value during a slow SOCKS5 handshake, and contrary to the intention, copy the too long hostname to the target buffer instead of copying just the resolved address there.

CVSS v3.1 Base Score	8.8
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H/E:P/RL:O/RC:C
CWE	CWE-122: Heap-based Buffer Overflow

Vulnerability CVE-2023-38546

This flaw allows an attacker to insert cookies at will into a running program using libcurl, if the specific series of conditions are met.

libcurl performs transfers. In its API, an application creates "easy handles" that are the individual handles for single transfers.

libcurl provides a function call that duplicates en easy handle called curl_easy_duphandle.

If a transfer has cookies enabled when the handle is duplicated, the cookie-enable state is also cloned - but without cloning the actual cookies. If the source handle did not read any cookies from a specific file on disk, the cloned version of the handle would instead store the file name as `none` (using the four ASCII letters, no quotes).

Subsequent use of the cloned handle that does not explicitly set a source to load cookies from would then inadvertently load cookies from a file named `none` - if such a file exists and is readable in the current directory of the program using libcurl. And if using the correct file format of course.

CVSS v3.1 Base Score	3.7
CVSS Vector	CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:N/E:P/RL:O/RC:C
CWE	CWE-73: External Control of File Name or Path

Vulnerability CVE-2023-46218

This flaw allows a malicious HTTP server to set "super cookies" in curl that are then passed back to more origins than what is otherwise allowed or possible. This allows a site to set cookies that then would get sent to different and unrelated sites and domains. It could do this by exploiting a mixed case flaw in curl's function that verifies a given cookie domain against the Public Suffix List (PSL). For example a cookie could be set with `domain=co.UK` when the URL used a lower case hostname `curl.co.uk`, even though `co.uk` is listed as a PSL domain.

CVSS v3.1 Base Score	6.5
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C
CWE	CWE-20: Improper Input Validation

Vulnerability CVE-2023-46219

When saving HSTS data to an excessively long file name, curl could end up removing all contents, making subsequent requests using that file unaware of the HSTS status they should otherwise use.

CVSS v3.1 Base Score	5.3
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:N/E:P/RL:O/RC:C
CWE	CWE-311: Missing Encryption of Sensitive Data

Vulnerability CVE-2024-30206

Affected SIMATIC RTLS Locating Manager Clients do not properly check the integrity of update files. This could allow an unauthenticated remote attacker to alter update files in transit and trick an authorized user into installing malicious code. A successful exploit requires the attacker to be able to modify the communication between server and client on the network.

CVSS v3.1 Base Score	8.8
CVSS Vector	CVSS:3.1/AV:A/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:H/E:P/RL:O/RC:C
CVSS v4.0 Base Score	8.8
CVSS Vector	CVSS:4.0/AV:A/AC:L/AT:P/PR:N/UI:A/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H
CWE	CWE-494: Download of Code Without Integrity Check

Vulnerability CVE-2024-30207

The affected systems use symmetric cryptography with a hard-coded key to protect the communication between client and server. This could allow an unauthenticated remote attacker to compromise confidentiality and integrity of the communication and, subsequently, availability of the system. A successful exploit requires the attacker to gain knowledge of the hard-coded key and to be able to intercept the communication between client and server on the network.

CVSS v3.1 Base Score	10.0
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H/E:P/RL:O/RC:C
CVSS v4.0 Base Score	10.0
CVSS Vector	CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H
CWE	CWE-321: Use of Hard-coded Cryptographic Key

Vulnerability CVE-2024-30208

The “DBTest” tool of SIMATIC RTLS Locating Manager does not properly enforce access restriction. This could allow an authenticated local attacker to extract sensitive information from memory.

CVSS v3.1 Base Score	6.3
CVSS Vector	CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:C/C:L/I:L/A:L/E:P/RL:O/RC:C
CVSS v4.0 Base Score	5.2
CVSS Vector	CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:L/VI:L/VA:L/SC:H/SI:H/SA:H
CWE	CWE-732: Incorrect Permission Assignment for Critical Resource

Vulnerability CVE-2024-30209

Affected systems transmit client-side resources without proper cryptographic protection. This could allow an attacker to eavesdrop on and modify resources in transit. A successful exploit requires an attacker to be in the network path between the RTLS Locating Manager server and a client (MitM).

CVSS v3.1 Base Score	9.6
CVSS Vector	CVSS:3.1/AV:A/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H/E:P/RL:O/RC:C
CVSS v4.0 Base Score	9.0
CVSS Vector	CVSS:4.0/AV:A/AC:L/AT:P/PR:N/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H
CWE	CWE-319: Cleartext Transmission of Sensitive Information

Vulnerability CVE-2024-33494

Affected components do not properly authenticate heartbeat messages. This could allow an unauthenticated remote attacker to affected the availability of secondary RTLS systems configured using a TeeRevProxy service and potentially cause loss of data generated during the time the attack is ongoing.

CVSS v3.1 Base Score	6.5
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:L/E:P/RL:O/RC:C
CVSS v4.0 Base Score	6.9
CVSS Vector	CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:L/VA:L/SC:N/SI:N/SA:N
CWE	CWE-345: Insufficient Verification of Data Authenticity

Vulnerability CVE-2024-33495

The affected application does not properly limit the size of specific logs. This could allow an unauthenticated remote attacker to exhaust system resources by creating a great number of log entries which could potentially lead to a denial of service condition. A successful exploitation requires the attacker to have access to specific SIMATIC RTLS Locating Manager Clients in the deployment.

CVSS v3.1 Base Score	6.5
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H/E:P/RL:O/RC:C
CVSS v4.0 Base Score	7.1
CVSS Vector	CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N
CWE	CWE-770: Allocation of Resources Without Limits or Throttling

Vulnerability CVE-2024-33496

Affected SIMATIC RTLS Locating Manager Report Clients do not properly protect credentials that are used to authenticate to the server. This could allow an authenticated local attacker to extract the credentials and use them to escalate their access rights from the Manager to the Systemadministrator role.

CVSS v3.1 Base Score	6.3
CVSS Vector	CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:C/C:L/I:L/A:L/E:P/RL:O/RC:C
CVSS v4.0 Base Score	4.8
CVSS Vector	CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:L/VI:L/VA:L/SC:L/SI:L/SA:L
CWE	CWE-522: Insufficiently Protected Credentials

Vulnerability CVE-2024-33497

Affected SIMATIC RTLS Locating Manager Track Viewer Client do not properly protect credentials that are used to authenticate to the server. This could allow an authenticated local attacker to extract the credentials and use them to escalate their access rights from the Manager to the Systemadministrator role.

CVSS v3.1 Base Score	6.3
CVSS Vector	CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:C/C:L/I:L/A:L/E:P/RL:O/RC:C
CVSS v4.0 Base Score	4.8
CVSS Vector	CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:L/VI:L/VA:L/SC:L/SI:L/SA:L
CWE	CWE-522: Insufficiently Protected Credentials

Vulnerability CVE-2024-33498

Affected applications do not properly release memory that is allocated when handling specifically crafted incoming packets. This could allow an unauthenticated remote attacker to cause a denial of service condition by crashing the service when it runs out of memory. The service is restarted automatically after a short time.

CVSS v3.1 Base Score	5.3
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L/E:P/RL:O/RC:C
CVSS v4.0 Base Score	6.9
CVSS Vector	CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N
CWE	CWE-400: Uncontrolled Resource Consumption

Vulnerability CVE-2024-33499

The affected application assigns incorrect permissions to a user management component. This could allow a privileged attacker to escalate their privileges from the Administrators group to the Systemadministrator group.

CVSS v3.1 Base Score	9.1
CVSS Vector	CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:C/C:H/I:H/A:H/E:P/RL:O/RC:C
CVSS v4.0 Base Score	9.4
CVSS Vector	CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H
CWE	CWE-732: Incorrect Permission Assignment for Critical Resource

Vulnerability CVE-2024-33583

Affected application contains a hidden configuration item to enable debug functionality. This could allow an authenticated local attacker to gain insight into the internal configuration of the deployment.

CVSS v3.1 Base Score	3.3
CVSS Vector	CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N/E:P/RL:O/RC:C
CVSS v4.0 Base Score	4.8
CVSS Vector	CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N
CWE	CWE-912: Hidden Functionality

ADDITIONAL INFORMATION

For further inquiries on security vulnerabilities in Siemens products and solutions, please contact the Siemens ProductCERT:

https://www.siemens.com/cert/advisories

HISTORY DATA

V1.0 (2024-05-14): Publication Date
V1.1 (2024-06-11): Added specific mitigation for CVE-2024-30207

TERMS OF USE