C04-12.2.1

Liaison Note to IEC TC80

S-100 vulnerability

1 INTRODUCTION

DTEC discussed the security vulnerability that has been pointed out in the article 'Analysis of Security Vulnerabilities in S-100-Based Maritime Navigation Software'[1] in the publication 'Cyber Physical System: Security and Resilience Challenges and Solutions'.[2] The vulnerability relates to the use of the Lua scripting language for portrayal in S-100.

Apart from this liaison note the information herein has also been provided to IHO in a separate liaison note.

2 DISCUSSION

2.1 Overview of vulnerability

S-100 Part 9a describes using scripting for portrayal of S-100 products as defined in S-100 Part 13. The scripting functionality uses the Lua programming language version 5.1. The security vulnerability is very severe (Common Vulnerability Scoring System (CVSS) score 9.3) that can enable remote code execution through malicious scripts. As S-100 does not limit what libraries are allowed when using scripting for portrayal and there are no sandboxing requirements the threat is valid and severe for all consumers of S-100 products and not just the ECDIS.

Further investigation and literature review during DTEC6 has noted that support for Lua 5.1 has ended in February 2012 and there are still open vulnerabilities found in Lua 5.1 that have not been and will not be fixed that can cause Denial of Service attacks on systems[3]. The currently maintained version of Lua is 5.5 with support and updates still possible for 5.4.

2.2 Impact

As the vulnerability requires the S-100 product to be delivered in an ExchangeSet and by definition ExchangeSets must include a signature, accepting only ExchangeSets from trusted sources reduces the impact to some degree. However, as in all cyber security implicitly trusting a source without verifying input is not considered a good practice and all inputs must be validated and verified. Outside of the maritime domain there have been multiple instances of state-sponsored cyber-attacks which cannot be left outside of consideration in the maritime domain and state-sponsored actors may either have access to valid and trusted certificates or be able to place agent programmers in positions where the software is created.

The impact of the currently open CVEs in Lua 5.1 is mostly limited to Denial of Service (i.e. crashes) of the running system. Depending on the system and how it operates, and installation of a malicious or malformed Lua script that causes such a crash may prevent restarting the system and lead to a crash loop until the script causing the crash can be identified and removed from the system.

[1] Available at https://www.mdpi.com/1424-8220/26/4/1246
[2] Hoyeon Cho, Division of Maritime Information Technology, National Korea Maritime and Ocean University, Busan 49112, Republic of Korea; Changui Lee, SemanticWave, Busan 47257, Republic of Korea; Seojeong Lee, Division of Marine System Engineering, National Korea Maritime and Ocean University, Busan 49112, Republic of Korea
[3] See https://app.opencve.io/cve/?vendor=lua

The impact of the remote code execution vulnerability that is not dependent on the Lua version is more severe as it can lead to the attacker gaining access to the underlying system. This can lead to installation of other backdoors etc that are no longer dependent on the scripting environment or the original system to be running.

2.3 Mitigation options

The mitigation options depend on the use case of the S-100 product, but the following options may be considered:

2.3.1 Data validation and sanitation

Ensuring that the received data comes from a trusted source via certificate and signature checks is the first step. However, blindly trusting incoming scripts from even a trusted source is not valid approach and input sanitation must also be considered.

For use cases in which portrayal is not required to be received (e.g. bi-directional route transfer in S-421 or distribution of navigational warnings in S-124) recipients of the data can and should interpret a received ExchangeSet with Lua scripting present as malicious data and reject it outright.

2.3.2 Sandboxing script execution

As suggested in the research paper, sandboxing the script execution to an environment that does not allow access to the underlying system and in which the crash of the sandboxed environment does not cause a crash of the application itself will allow for safer execution of the script. The performance implication of sandboxing must be evaluated separately.

There is a performant and sandboxed derivative of Lua 5.1 called Luau[4] that may prove to be an adequate sandboxing environment for S-100 use but this needs to be evaluated further.

2.3.3 Limiting allowed libraries

S-100 consumers could limit the allowed libraries that Lua scripts can utilize to provide a more secure approach without the need for sandboxing (see the previous footnote). However, this does not solve the open issues with Lua 5.1 open CVEs that are not going to be fixed.

2.3.4 Updating Lua

As Lua 5.1 has not received any updates since 2012 and there are open Common Vulnerabilities and Exposures (CVEs) it is recommended that a more modern Lua runtime is used in combination with the previous mitigation strategies. As Lua 5.5 is not necessarily backwards compatible to Lua 5.1 this may cause issues with existing portrayals.

3 ACTION REQUESTED

IEC TC80 is requested to

- take note of the information
- consider how it applies to information systems in their domain that either generate or consume S-100 products
- keep IALA and other stakeholders informed of other security vulnerabilities that are found in S-100 and related technologies

[4] https://luau.org/sandbox/