Ada Web Server does not use Crypto Secure Pseudo Random Number Generator

ADACORE SECURITY ADVISORY

Document Id: SEC.AWS-0040

Version 2 - Jul 31, 2024

AdaCore

Title	Ada Web Server does not use Crypto Secure Pseudo Random Number Generator	
Status	Final	
Author	Frederic Leger	
Reviewed by	Pascal Obry, Olivier Ramonat, Johannes Kliemann	

Revision History

Version	Date	Comments
2	Jul 30, 2024	Issue is fixed on wavefront
1	Jul 16, 2024	Initial version

Contents

1. Preface

1.1. Scope

This document is an advisory describing the security impact of `AWS-0040`. The issue is tracked under the ticket number `AWS-0040` in AdaCore's issue tracking database.

This document also presents possible workarounds and mitigations for the issue.

1.2. Distribution

This advisory is made available in confidence to AdaCore customers under embargo until 2024-09-23 so that they can address the issue it describes before public availability.

Thereafter, it will be available to the general public under the terms of the `CC BY-ND 4.0` licence.

1.3. Contact

For questions on this document, please contact AdaCore support at product-security@adacore.com or using the standard reporting procedures if you are an AdaCore customer.

2. Vulnerability

2.1. Affected Products

The vulnerability described in this document was reported for the following product versions:

- *Ada Web Server* - `aws` *20.00* (20191009)

The vulnerability impacts the generated random values within the `aws` modules:

- The Random Number Generator of Ada is not cryptographically secure, and some of `aws` random values **should** use a Cryptographically Secure Pseudo Random Numbers Generator.
- The implementation of the `Random_String()` function in the `src/core/aws-utils.adb` module introduces a bias in the generated random string values, where the values `"1"` and `"2"` have a much higher frequency than any other character.

The vulnerability described in this document applies to the following product versions:

- *Ada Web Server* - `aws` < 25.1

2.2. Severity and Impact

CVSS v3.1 score: 7.5 (AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N/E:U/RL:U/RC:C)

This issue can possibly cause a *session hijack* attack on an *Ada Web Server* if an attacker extracts sufficient PRNG data from legitimate requests, the secret state of the Meresenne Twister PRNG may be retrieved.

A CVE (*Common Vulnerabilities and Exposures*) has been created for that case, and will be referred as `CVE-2024-41708`.

2.3. Detailed Description

Thanks to another great investigation work by Chris Culnane and Ty Wilson-Brown a vulnerability on the the *Ada Web Server* has been discovered.

The implementation of the `Random_String()` function in the `src/core/aws-utils.adb` module introduces a bias in the generated random string values, where the values `"1"` and `"2"` have a much higher frequency than any other character.

Moreover, the `Numerics.Discrete_Random` module is used for random numbers generation, but the PRNG is not Cryptographically Secure. At least for the session ID and the session private key generation, a CSPRNG (Cryptographically Secure Pseudo Random Numbers Generator) should be used.

The internal state of the Meresenne Twister PRNG may be retrieved, and lead to a session hijacking attack.

3. Solution

3.1. Workarounds

There is no workaround.

3.2. Correction

The vulnerability described in this document is corrected in the following product versions:

- *Ada Web Server* - `aws` > 25.0
- *GNAT Components Collection - Core packages* - `gnatcoll` > 25.0

Starting from build date `20240717`, the *Ada Web Server* component is fixed on `wavefront`. It also requires the update of *GNAT Components Collection - Core packages*, which contains a new CSPRNG package named `GNATCOLL.Random`.

4. Appendix

4.1. CVSS Score Justification

Metric	Justification
AV:N	A vulnerability exploitable with Network access means the vulnerable component is bound to the network stack and the attacker's path is through OSI layer 3 (the network layer). Such a vulnerability is often termed 'remotely exploitable' and can be thought of as an attack being exploitable one or more network hops away (e.g. across layer 3 boundaries from routers).
AC:L	Specialized access conditions or extenuating circumstances do not exist. An attacker can expect repeatable success against the vulnerable component.
PR:N	The attacker is unauthorized prior to attack, and therefore does not require any access to settings or files to carry out an attack.
UI:N	The vulnerable system can be exploited without interaction from any user.
S:U	An exploited vulnerability can only affect resources managed by the same authority. In this case the vulnerable component and the impacted component are the same.
C:H	There is total loss of confidentiality, resulting in all resources within the impacted component being divulged to the attacker. Alternatively, access to only some restricted information is obtained, but the disclosed information presents a direct, serious impact.
I:H	There is no loss of integrity within the impacted component.
A:N	There is no impact to availability within the impacted component.
E:U	No exploit code is available, or an exploit is entirely theoretical.
RL:U	There is either no solution available or it is impossible to apply.
RC:C	Detailed reports exist, or functional reproduction is possible (functional exploits may provide this). Source code is available to independently verify the assertions of the research, or the author or vendor of the affected code has confirmed the presence of the vulnerability.