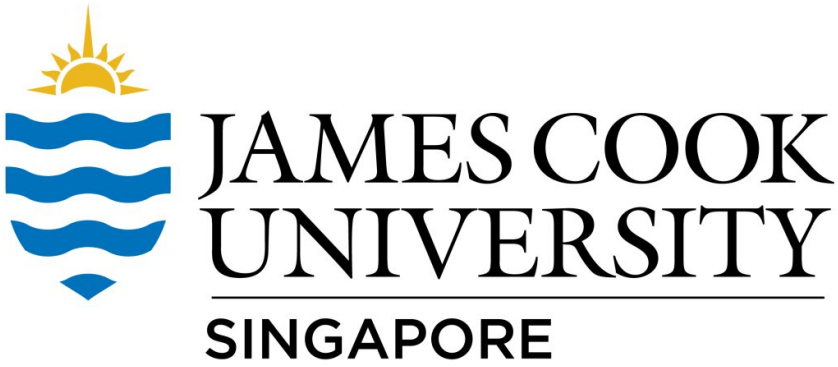


CP2422

Cloud and Data Center Security

Pyae Sone Soe Moe

Table of Contents

Introduction

The main developments and security upgrades in WordPress, TLS security, cryptographic methods, certificates, and SQL databases are described in this document. These enhancements strengthen overall efficiency while enhancing dependability and security of key technologies by reducing vulnerabilities.

Vulnerabilities have been fixed in WordPress, making it a more effective and secure platform. The security of data transmission has been improved with the update of TLS security from version 1.2 to 1.3. ECDSA has replaced RSA as the primary cryptographic algorithm, enhancing security during dynamic testing. Improvements to certificate handling guarantee proper application in dynamic tests. The latest version of SQL databases, 8.0.31, offer enhanced performance, security, and compatibility. Together, these improvements improve the technical environment.

Wordpress

The current WordPress version in use has flaws that leave it open to different kinds of attacks, including Cross-Site Scripting Vulnerabilities and Zero-Day Vulnerabilities. The services that rely on these applications may suffer as a result of these vulnerabilities, which present serious dangers.

```
+   # NOTE: Changing versions AFTER deploying m
+   # if an upgrade fails, or you try to downgr
+   # Sometimes, returning to the previous veri
+   # Worst case: stop and redeploy your enviro
+   #
+   WORDPRESS_VERSION: 5.9.0-php8.1
+   MYSQL_VERSION: 8.0.29
+
+   #
+   # Don't change these variables as they're r
+   #
+   KUBE_NAMESPACE: ${CI_PROJECT_ROOT_NAMESPACE
```

The changes made to the code have produced an updated version of WordPress that naturally reduces the previously mentioned vulnerabilities. Programs utilizing this upgraded version are anticipated to function more effectively and securely as a result of these modifications, lowering the likelihood of the aforementioned vulnerabilities.

```
    # if an upgrade fails, or you try to downgr
    # Sometimes, returning to the previous veri
    # Worst case: stop and redeploy your enviro
    #
    WORDPRESS_VERSION: 6.3.0-php8.2
    MYSQL_VERSION: 8.1.0

    #
    # Don't change these variables as they're n
    #
    KUBE_NAMESPACE: ${CI_PROJECT_ROOT_NAMESPACE
    KUBE_CONTEXT: ${CI_PROJECT_ROOT_NAMESPACE}/
```

https://gitlab.bcyber.online/cp2422/2023sp52jcus/pyae/blog-webapp/-/commit/f624d8dc043fae66c8cb33dde05348b959d9c6d5

TLS Version

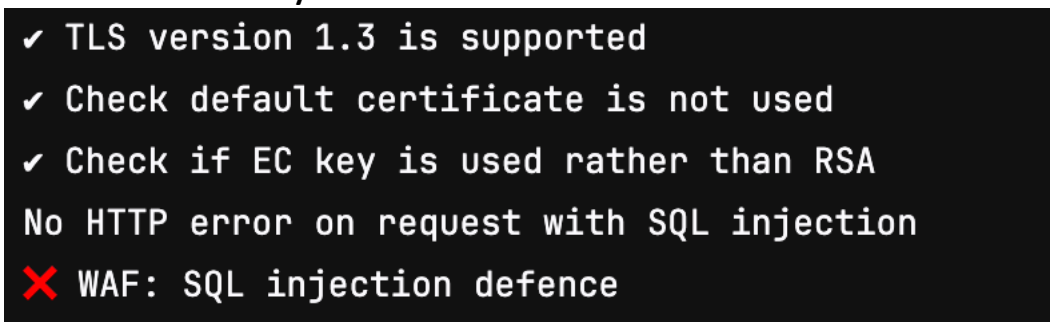
The Transport Layer Security (TLS) version had been set to utilize TLS version 1.2 in the traefik-ingress.yaml file. The dynamic test failed as a result of this particular configuration.

```
HTTP 302: Temporary redirect. Installation is prob
❌ Port 80 (HTTP) should error or redirect to HTTP
❌ TLS version 1.3 is supported
❌ Check default certificate is not used
❌ Check if EC key is used rather than RSA
```

The maximum Transport Layer Security (TLS) version was changed in the traefik-ingress.yaml file, specifically moving it from 1.2 to 1.3. These modifications were then saved and documented as commits.

```yaml
apiVersion: traefik.containo.us/v1alpha1
kind: TLSOption
metadata:
  name: tlsoptions
spec:
  maxVersion: VersionTLS13
  minVersion: VersionTLS11
```

TLS version 1.3 was an improvement and update to the original Transport Layer Security (TLS) version. The result of this particular upgrade was that the dynamic test could be run successfully.

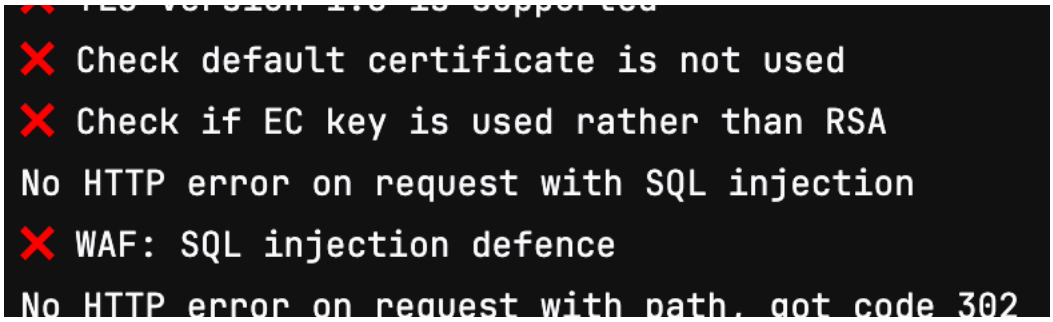


https://gitlab.bcyber.online/cp2422/2023sp52jcus/pyae/blog-webapp/-/commit/f589ce4abcd30df74b1ca32b6beea41d4dfcfd94

Algorithm

The RSA with size 2048 was used in the certificate.yaml file, which regrettably resulted in a failure when the dynamic test

was executed. The presence of adequate and sufficient credentials is required for the test to operate as intended.



The cryptographic algorithm for encryption was changed from RSA (Rivest-Shamir-Adleman) to ECDSA (Elliptic Curve Digital Signature Algorithm) in the certificate.yaml file at line 16. The key size for the algorithm was also adjusted at line 17 from 2048 bits to 384 bits, and these changes were then saved and committed to the file.

```
# This domain variable will automatically be set by
commonName: ${DOMAIN}
privateKey:
  algorithm: ECDSA
  size: 384
usages:
  - server auth
```

Using the changes made to the certificate.yaml file, the appropriate cryptographic algorithm was used. The dynamic test was therefore successfully completed, demonstrating that the EC (Elliptic Curve) key was used in place of the RSA (Rivest-Shamir-Adleman) method.

```
✔ TLS version 1.3 is supported
✔ Check default certificate is not used
✔ Check if EC key is used rather than RSA
No HTTP error on request with SQL injection
❌ WAF: SQL injection defence
```

https://gitlab.bcyber.online/cp2422/2023sp52jcus/pyae/blog-webapp/-/commit/0eac3e5a6d5e528d85eda9e1509ae2392e69a293

https://gitlab.bcyber.online/cp2422/2023sp52jcus/pyae/blog-webapp/-/commit/10b2ad17bead4fd03986d84b7f3af33b3fbb7445

Certificate

The default certificate was used in the kustomization.yaml file because the essential code to implement the right certificates had been commented out. As a result, it was discovered that the default certificate was being used throughout the dynamic test.

```
+ resources:
+   - mysql-deployment.yaml
+   - wordpress-deployment.yaml
+   - traefik-ingress.yaml
+   - pki.yaml
+   # Not used _right now_, but maybe useful to have?
+   #- certificate.yaml
+   #- waf.yaml
```

To ensure that the code would work properly with the necessary certifications, the pertinent areas of code were purposefully commented out. This was accomplished by integrating the waf.yaml and certification.yaml files, making the required adjustments, and then committing the code changes.

kustomization.yaml 191 B

```yaml
resources:
  - mysql-deployment.yaml
  - wordpress-deployment.yaml
  - traefik-ingress.yaml
  - pki.yaml
  # Not used _right now_, but maybe useful to have?
  - certificate.yaml
  - waf.yaml
```

https://gitlab.bcyber.online/cp2422/2023sp52jcus/pyae/blog-webapp/-/commit/497548fbbff4c1581c89ec40cfdce54ab76cb900

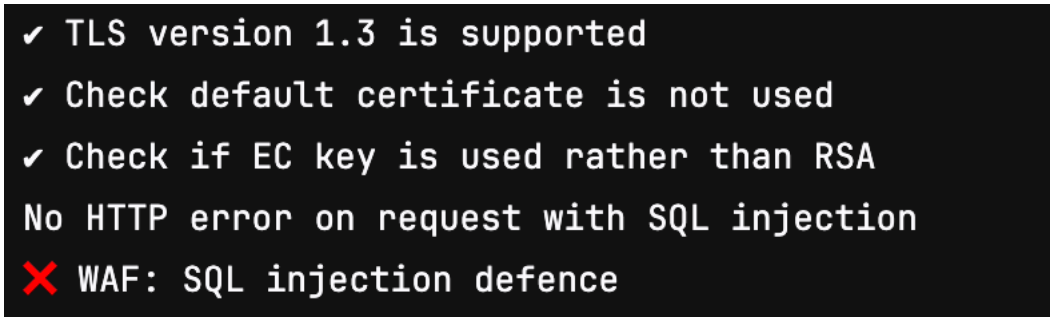
To guarantee that the Public Key Infrastructure (PKI) operates correctly, particular code pertaining to the TLS certificate secret name was purposefully left out of the wordpress-deployment.yaml file. However, as a result of this move, the dynamic test identified the use of the default certification, leading to a failure.

```
+      port: 8080
+  tls:
+    # TODO: We'd need this to be created by PKI in order to use it
+    # So the below line is commented out for now...
+    # secretName: wordpress-tls-secret
+    options:
+      name: tlsoptions
+    domains:
+    - main: ${DOMAIN}
```

The code responsible for TLS certification that was previously commented out was uncommented to make sure the code executes with the correct certificate. This step was made to stop the code from using the default certification during execution.

```
      name: wordpress
      port: 8080
  tls:
    # TODO: We'd need this to be created by PKI in order to use it
    # So the below line is commented out for now...
    secretName: wordpress-tls-secret
    options:
      name: tlsoptions
    domains:
    - main: ${DOMAIN}
---
```

After included all the dependencies and making the necessary code modifications, the software ran correctly without ever detecting any default certificates. As a consequence, it successfully completed the dynamic test.



https://gitlab.bcyber.online/cp2422/2023sp52jcus/pyae/blog-webapp/-/commit/3b34ffbc2a116a15b88baaf2a06e13f05c03ffcf

Database

The current SQL version has been identified to contain security weaknesses that could be exploited by bad actors to carry out damaging attacks. Therefore, it is crucial to start making improvements and security improvements to address these vulnerabilities and improve the system's overall security.

```
+   # NOTE: Changing versions AFTER deploying m
+   # if an upgrade fails, or you try to downgr
+   # Sometimes, returning to the previous veri
+   # Worst case: stop and redeploy your envirc
+   #
+   WORDPRESS_VERSION: 5.9.0-php8.1
+   MYSQL_VERSION: 8.0.29
+
+   #
+   # Don't change these variables as they're r
+   #
+   KUBE_NAMESPACE: ${CI_PROJECT_ROOT_NAMESPACE
```

Version 8.0.29 of the SQL database management system has been upgraded to version 8.0.31, which is more modern. The most current upgrade has a number of benefits, including improved performance capabilities for quicker and more efficient operations, strengthened security measures to guard against potential vulnerabilities, and the correction of known concerns through bug fixes. Additionally, it adds compatibility features that guarantee seamless integration with other programs and systems, making the environment for databases run more smoothly and dependably. New version can bring benefits such as security enhancements, improved performance along with bug fixes and compatibility features.

```
3    # Sometimes, returning to the previous ver
4    # Worst case: stop and redeploy your envir
5    #
6    WORDPRESS_VERSION: 6.3.0-php8.2
7    MYSQL_VERSION: 8.0.31
8
9    #
0    # Don't change these variables as they're
```

https://gitlab.bcyber.online/cp2422/2023sp52jcus/pyae/blog-webapp/-/commit/540e7ba7e8e791abcb714dd1ba60d89795cf740c

Conclusion

There is a transition from the antiquated TLS 1.2 to the strong and secure TLS 1.3 in the broader context of cybersecurity. While boosting data transfer security dramatically, this shift ensures compatibility. Additionally, the transition from the RSA algorithm to the robust ECDSA encryption, along with key size modifications, strengthens security during dynamic testing in the context of cryptography. To ensure their correct application and success in dynamic testing scenarios, certificates essential for secure communication undergo strategic adjustments and integrations. The SQL database also advances from 8.0.29 to 8.0.31, which offers improved speed, increased security, bug fixes, and compatibility improvements. Together, these changes strengthen and secure the technical environment.