securitum

Security report

SUBJECT

Security test of [CLIENT_NAME] WAN infrastructure

DATE

27.06.2024 – 01.07.2024

RETEST DATE

22.10.2024

LOCATION

Poznan (Poland)

AUTHOR

Paweł Różański

VERSION

1.1

securitum

Executive summary

This document is a summary of work conducted by the Securitum. The subject of the test was WAN cloud infrastructure available at the following IP addresses:

- [IP_ADDERS1],
- [IP_ADDERS2],
- [IP_ADDERS3].

Tests were conducted as an unauthenticated user. To perform the test, exception was added on the firewall to permit traffic from Securitum IP. Before adding an exception on the firewall, all TCP ports on tested machines were blocked.

The most severe vulnerability identified during the assessment was:

- SECURITUM-ABXXXXXX-001: Unrestricted access to sensitive data in Consul's key/value store.

During the tests, particular emphasis was placed on vulnerabilities that might in a negative way affect confidentiality, integrity or availability of processed data.

The security tests were carried out according to generally accepted methodologies, as well as internal good practices of conducting security tests developed by the Securitum.

An approach based on manual tests supported by several automatic tools (i.a. Burp Suite Professional, Nessus Professional, THC hydra, dirsearch, ssh-audit, nmap), was used during the assessment.

The vulnerabilities are described in detail in further parts of the report.

Retest

Retest of following found vulnerabilities was performed:

- SECURITUM-ABXXXXXX-001,
- SECURITUM-ABXXXXXX-002,
- SECURITUM-ABXXXXXX-003,
- SECURITUM-ABXXXXXX-004,
- SECURITUM-ABXXXXXX-005.

Due to the nature of the infrastructure, which runs in the cloud, new IP addresses were indicated by the ordering party as equivalent to those previously tested:

- [REDACTED],
- [REDACTED],
- [REDACTED].

As during the original test, an exception was added to the firewall to permit traffic from Securitum IPs.

Two vulnerabilities were fixed, and one was partially fixed. Recommendations for informational issues were not implemented. Details are described in further parts of the report.

Risk classification

Vulnerabilities are classified on a five-point scale, that reflects both the probability of exploitation of the vulnerability and the business risk of its exploitation. Below, there is a short description of the meaning of each of the severity levels:

- CRITICAL – exploitation of the vulnerability makes it possible to compromise the server or network device, or makes it possible to access (in read and/or write mode) data with a high degree of confidentiality and significance. The exploitation is usually straightforward, i.e. an attacker does not need to gain access to the systems that are difficult to reach and does not need to perform social engineering. Vulnerabilities marked as 'CRITICAL' must be fixed without delay, mainly if they occur in the production environment.

- HIGH – exploitation of the vulnerability makes it possible to access sensitive data (similar to the 'CRITICAL' level), however the prerequisites for the attack (e.g. possession of a user account in an internal system) make it slightly less likely. Alternatively, the vulnerability is easy to exploit, but the effects are somehow limited.

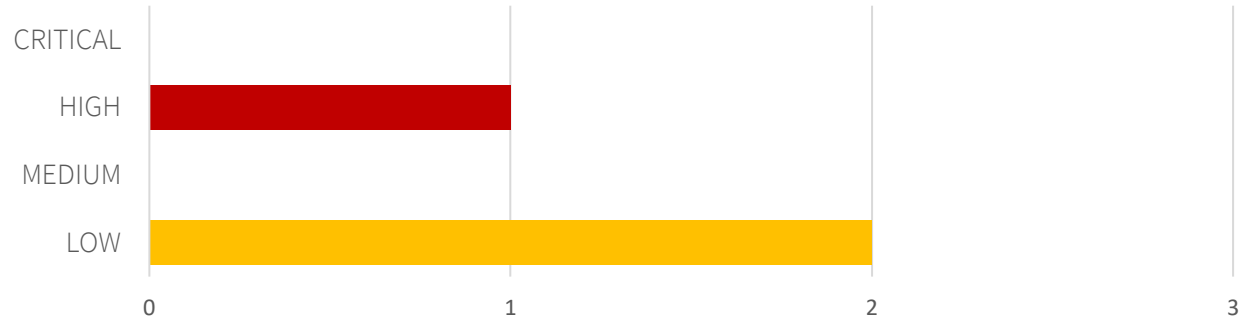
- MEDIUM – exploitation of the vulnerability might depend on external factors (e.g. convincing the user to click on a hyperlink) or other conditions that are difficult to achieve. Furthermore, exploitation of the vulnerability usually allows access only to a limited set of data or to data of a lesser degree of significance.

- LOW – exploitation of the vulnerability results in minor direct impact on the security of the test subject or depends on conditions that are very difficult to achieve in practical manner (e.g. physical access to the server).

- INFO – issues marked as 'INFO' are not security vulnerabilities per se. They aim to point out good practices, the implementation of which will lead to the overall increase of the system security level. Alternatively, the issues point out some solutions in the system (e.g. from an architectural perspective) that might limit the negative effects of other vulnerabilities.

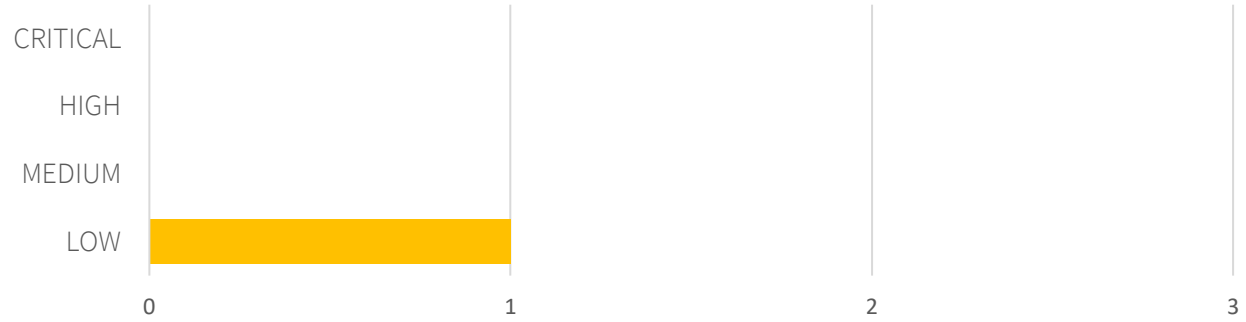
Statistical overview

Below, a statistical summary of vulnerabilities is shown:



Additionally, two INFO issues are reported.

Below, a statistical summary of vulnerabilities after retest on 22.10.2024 is shown:



Additionally, two INFO issues are reported.

Contents

Change history

Document date	Version	Change description
23.10.2024	1.1	Retest of all previously found vulnerabilities: • SECURITUM-ABXXXXXX-001, • SECURITUM-ABXXXXXX-002, • SECURITUM-ABXXXXXX-003, • SECURITUM-ABXXXXXX-004, • SECURITUM-ABXXXXXX-005.
01.07.2024	1.0	Creation of a document.

Vulnerabilities in the cloud infrastructure

[FIXED][HIGH] SECURITUM-ABXXXXXX-001: Unrestricted access to sensitive data in Consul's key/value store

STATUS AFTER RETEST

The audit showed that vulnerability was eliminated. During the retest connection to port 8500 TCP was not possible. It was confirmed with ordering party that Consul service, which was using that port, was entirely removed.

SUMMARY

During audit it was observed that access to Conul's key/value data is possible without authorization. Data in key/value store contain configuration of services, including sensitive data like secrets, passwords, access keys. Such configuration can lead to compromising of all secrets in key/value store in case of exploitation of any other vulnerability (i.e. SSRF, https://owasp.org/www-community/attacks/Server_Side_Request_Forgery) in single service.

Write and delete operations were not tested as test was performed on production environment.

Severity of this vulnerability was lowered from CRITICAL as with initial configuration access to this port is restricted by firewall.

PREREQUISITES FOR THE ATTACK

Access to port 8500 TCP by attacker.

TECHNICAL DETAILS (PROOF OF CONCEPT)

Getting key list:

Request:

```
GET /v1/kv/?keys HTTP/1.1
Host: [IP_ADDERS2]:8500
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json
[…]

[
  "alertmanager/alertmanager.yml",
  "configs/global/amqp/deadLetterExchange",
  "configs/global/amqp/deadLetterQueue",
  "configs/global/amqp/maxRetries",
  "configs/global/amqp/queueTTL",
  "configs/global/amqp/reconnectAttempts",
  "configs/global/amqp/reconnectTimeout",
  "configs/global/amqp/url",
  "configs/global/aws/config/accessKeyId",
  "configs/global/aws/config/region",
  "configs/global/aws/config/secretAccessKey",
[…]
```

```
  "configs/services/collaboration-statistics/database/password",
  "configs/services/collaboration-statistics/database/ssl",
  "configs/services/collaboration-statistics/database/user",
[…]
  "configs/services/users/database/password",
  "configs/services/users/database/ssl",
  "configs/services/users/database/user",
```

Getting example value:

Request:

```
GET /v1/kv/configs/services/comment-threads/database/username HTTP/1.1
Host: [IP_ADDERS2]:8500
```

Response (censored):

```
HTTP/1.1 200 OK
Content-Type: application/json
[…]

[{"LockIndex":0,"Key":"configs/services/comment-
threads/database/username","Flags":0,"Value":"Y********=",[…]}]
```

LOCATION

[REDACTED]

RECOMMENDATION

It is recommended to use dedicated system, Vault, to store secrets and sensitive data securely. Consul's key/value storage security still can be improved by using built-in mechanism such as mTLS or ACLs, according to official Consul's documentation.

More information:

- https://developer.hashicorp.com/consul/tutorials/production-vms/security,
- https://developer.hashicorp.com/vault.

[PARTIALLY FIXED][LOW] SECURITUM-ABXXXXXX-002: Outdated software

STATUS AFTER RETEST

The test showed that vulnerability was partially eliminated:

- OpenSSH was upgraded to version 8.7 which is not the latest version,
- Consul service was entirely removed,
- cAdvisor was upgraded to v0.49.1 which is the latest version:

```
cadvisor_version_info{cadvisorRevision="[…]",cadvisorVersion="v0.49.1",dockerVersion="",kernelVer
sion="6.1.102-108.177.amzn2023.x86_64",osVersion="Alpine Linux v3.18"} 1
```

Current outdated software status:

Example location / URL	Software and version
[REDACTED] port 22 TCP	OpenSSH 8.7
[REDACTED] port 22 TCP	OpenSSH 8.7
[REDACTED] port 22 TCP	OpenSSH 8.7

SUMMARY

It was observed that software components are not updated to the newest version, and it can be found that it contains publicly known vulnerabilities.

Components identified in tested systems:

- OpenSSH 7.4,
- Consul 1.12.0,
- cAdvisor v0.44.0.

During the tests it was not possible to prepare a working Proof of Concept using the described vulnerability, however the mere fact of using software with publicly known vulnerabilities exhausts the necessity to include such information in the report.

More information:

- https://nvd.nist.gov/vuln/detail/CVE-2016-20012,
- https://nvd.nist.gov/vuln/detail/cve-2018-15473,
- https://nvd.nist.gov/vuln/detail/CVE-2018-15919,
- https://nvd.nist.gov/vuln/detail/CVE-2020-15778,
- https://nvd.nist.gov/vuln/detail/CVE-2021-41617,
- https://github.com/google/cadvisor/issues/3024,
- https://nvd.nist.gov/vuln/detail/CVE-2022-40716.

PREREQUISITES FOR THE ATTACK

Depends on vulnerability.

TECHNICAL DETAILS (PROOF OF CONCEPT)

Below, there is a table with the version of the current software along with the hosts which it has been identified on:

Example location / URL	Software and version
[REDACTED]	OpenSSH 7.4
[REDACTED]	OpenSSH 7.4
[REDACTED]	cAdvisor v0.44.0
[REDACTED]	cAdvisor v0.44.0
[REDACTED]	Consul 1.12.0
[REDACTED]	Consul 1.12.0
[REDACTED]	Consul 1.12.0

LOCATION

Locations included in the technical details section.

RECOMMENDATION

It is recommended to update the software to the latest, stable version.

[FIXED][LOW] SECURITUM-ABXXXXXX-003: Support for outdated SSH algorithms

STATUS AFTER RETEST

The retest showed that weak algorithms indicated after the initial test were removed.

SUMMARY

The tested hosts support weak SSH algorithms, which are used to set up a secure communication channel. This could pose a risk of compromising or modifying sensitive user data if an attacker eavesdrops network traffic (Man in the Middle attack, MITM).

More information:

- https://cwe.mitre.org/data/definitions/326.html
- https://cwe.mitre.org/data/definitions/327.html

PREREQUISITES FOR THE ATTACK

Performing a Man in the Middle attack.

TECHNICAL DETAILS (PROOF OF CONCEPT)

SSH servers supports the following deprecated key exchange algorithms:

```
diffie-hellman-group-exchange-sha1
diffie-hellman-group1-sha1
diffie-hellman-group14-sha1
```

SSH servers supports the following deprecated key encryption algorithms:

```
3des-cbc
cast128-cbc
blowfish-cbc
aes128-cbc
aes192-cbc
aes256-cbc
```

LOCATION

[REDACTED]

RECOMMENDATION

It is recommended to disable support for the algorithms mentioned above.

The current recommended OpenSSH configuration can be found at:

- https://infosec.mozilla.org/guidelines/openssh.html

Informational issues

[NOT IMPLEMENTED][INFO] SECURITUM-ABXXXXXX-004: cAdvisor exposed without authorization

STATUS AFTER RETEST

The retest showed that access to cAdvisor is still possible without authorization.

SUMMARY

During the audit, it was observed that access to cAdvisor application is available without form of authorization. This behaviour can help an attacker to gain information and better profile the environment, and use gained information to carry out further attacks.

Severity of this vulnerability was lowered to INFO as with initial configuration access to this port is restricted by firewall.

PREREQUISITES FOR THE ATTACK

Access to port 8088.

TECHNICAL DETAILS (PROOF OF CONCEPT)

Example of the HTTP request sent to the application:

```
GET /containers/ HTTP/1.1
Host: [IP_ADDERS1]:8088
```

In response, the application returns:

```
HTTP/1.1 200 OK
[…]

<html>
  <head>
    <title>cAdvisor - /</title>
```

LOCATION

[REDACTED]

RECOMMENDATION

It is recommended to protect access to cAdvisor with any form of authorization, even HTTP authentication with strong password.

More information:

- https://developer.mozilla.org/en-US/docs/Web/HTTP/Authentication

[NOT IMPLEMENTED][INFO] SECURITUM-ABXXXXXX-005: No protection of communication channel

STATUS AFTER RETEST

Retest showed that access to services is still possible without HTTPS.

SUMMARY

The tests have shown that the HTTPS protocol is not used to access the applications. This poses the risk of compromising or modifying sensitive user data if an attacker eavesdrops network traffic (Man in the Middle attack, MITM).

More information:

- https://cwe.mitre.org/data/definitions/757.html
- https://cwe.mitre.org/data/definitions/326.html

Severity of this vulnerability was lowered to INFO as presence of attacker in internal, backend network is required to perform the attack.

PREREQUISITES FOR THE ATTACK

Performing a Man in the Middle attack.

TECHNICAL DETAILS (PROOF OF CONCEPT)

The tested applications are available without HTTPS protocol:

LOCATION

All HTTP endpoints.

RECOMMENDATION

It is recommended to enforce the use of a secure, encrypted HTTPS communication channel. In addition, when trying to connect via HTTP, automatic redirection to HTTPS should be made.

The current recommended algorithm configuration can be found at:

- https://ssl-config.mozilla.org/

More information:

- https://cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Security_Cheat_Sheet.html