APPSECCO

CLIENT NAME –SAAS BASED WEB APPLICATION VULNERABILITY ASSESSMENT REPORT

Report for	CLIENT NAME CLIENT ADDRESS
Client contact	CLIENT CONTACT
Report title	CLIENT NAME –SaaS Based Web Application Vulnerability Assessment Report
Date	DD MONTH YYYY
Version	1.0.0
Status	FINAL
Appsecco contact	APPSECCO CONTACT

Version History

v1.0.0 – First version, current document.

APPSECCO

THE APPLICATION SECURITY COMPANY

Table of Contents

Version History 2

What this report contains 9

- Executive Summary 9
- Target Information 9
- Methodology 9
- Issue Summary 9
- Vulnerability by Severity 9
- Details of Individual Issues 9

Executive Summary 10

- Summary of Results 10
- Conclusion 12

Greybox Penetration Testing 13

- Target Information – SaaS based REDACTED Web Application 13
- Target Information – SaaS based REDACTED Internal Network 13
- Methodology 14
  - Testing Setup – SaaS based REDACTED Web Application 14
  - Testing Setup – SaaS based REDACTED Internal Network 14
- Approach 15
  - Greybox Penetration Testing - SaaS based REDACTED Internal Network 15
- Issue Summary 16
  - 1. Least privileged REDACTED can view Personally Identifiable Information (PII) of higher privileged users by accessing “REDACTED” 16
  - 2. An REDACTED (with no edit permissions) can create dashboard templates 16
  - 3. An REDACTED (with no edit permissions) can modify contents of any dashboard 16
  - 4. Least privileged users, like REDACTED, can read action logs of higher privileged users. 16
  - 5. Least privileged REDACTED can view personal details and log history of privileged users via IDOR in " REDACTED" 16
  - 6. It is possible to brute force a password for a known username using multiple credentials sets in the same request 16

7. Extra parameters in server response reveal sensitive information 16
8. Least privileged REDACTED can call " REDACTED" service to export any dashboard 16
9. Least privileged REDACTED can export and import any dashboard via vulnerable REST endpoints 17
10. IPs that are not part of subscription can be fetched via Insecure Direct Object Reference (IDOR) in “REDACTED” 17
11. REDACTED REST framework version is revealed via OPTIONS HTTP request method .. 17
Vulnerability Distribution by Severity 18
Summary of Issues by Severity 18
1. Least privileged REDACTED can view Personally Identifiable Information (PII) of higher privileged users by accessing “REDACTED” 19
Affected Assets 19
Severity 19
OWASP / CWE Mapping 19
Technical Description 19
Steps To Reproduce 19
Screenshots 19
Business Impact - So What? 20
Solution 20
Mitigation 20
References 21
2. An REDACTED (with no edit permissions) can create dashboard templates 22
Affected Assets 22
Severity 22
OWASP / CWE Mapping 22
Technical Description 22
Steps To Reproduce 22
Screenshots 22
Business Impact - So What? 23
Solution 23
Mitigation 23

References 23
3. An REDACTED (with no edit permissions) can modify contents of any dashboard 24
Affected Assets 24
Severity 24
OWASP / CWE Mapping 24
Technical Description 24
Steps To Reproduce 24
Screenshots 24
Business Impact - So What? 26
Solution 26
Mitigation 26
References 26
4. Least privileged users, like REDACTED, can read action logs of higher privileged users 27
Affected Assets 27
Severity 27
OWASP / CWE Mapping 27
Technical Description 27
Steps To Reproduce 27
Screenshots 27
Business Impact - So What? 27
Solution 28
Mitigation 28
References 28
5. Least privileged REDACTED can view personal details and log history of privileged users via IDOR in " REDACTED" 29
Affected Assets 29
Severity 29
OWASP / CWE Mapping 29
Technical Description 29
Steps To Reproduce 29
Screenshots 29

Business Impact - So What? 30
Solution 30
Mitigation 30
References 30
6. It is possible to brute force a password for a known username using multiple credentials sets in the same request 31
Affected Assets 31
Severity 31
OWASP / CWE Mapping 31
Technical Description 31
Steps To Reproduce 31
Screenshots 31
Business Impact - So What? 32
Solution 32
Mitigation 32
7. Extra parameters in server response reveal sensitive information 33
Affected Assets 33
Severity 33
OWASP / CWE Mapping 33
Technical Description 33
Steps To Reproduce 33
Screenshots 33
Business Impact - So What? 35
Solution 35
Mitigation 35
References 35
8. Least privileged REDACTED can call "REDACTED" service to export any dashboard 36
Affected Assets 36
Severity 36
OWASP / CWE Mapping 36
Technical Description 36

Steps To Reproduce 36
Screenshots 36
Business Impact - So What? 38
Solution 38
Mitigation 38
References 38
9. Least privileged REDACTED can export and import any dashboard via vulnerable REST endpoints 39
Affected Assets 39
Severity 39
OWASP / CWE Mapping 39
Technical Description 39
Steps To Reproduce 39
Screenshots 40
Business Impact - So What? 41
Solution 42
Mitigation 42
References 42
10. IPs that are not part of subscription can be fetched via Insecure Direct Object Reference (IDOR) in “REDACTED” 43
Affected Assets 43
Severity 43
OWASP / CWE Mapping 43
Technical Description 43
Steps To Reproduce 43
Screenshots 43
Business Impact - So What? 46
Solution 46
Mitigation 46
Recommendation 46
References 46

11. REDACTED REST framework version is revealed via OPTIONS HTTP request method 47
Affected Assets 47
Severity 47
OWASP / CWE Mapping 47
Technical Description 47
Steps To Reproduce 47
Screenshots 47
Business Impact - So What? 47
Solution 47
Recommendation 47
References 47
Conclusion 48

APPSECCO

THE APPLICATION SECURITY COMPANY

What this report contains

Executive Summary

This section provides a non-technical overview of the scope of the project, the key findings from the testing work carried out and highlights the areas we see as most important or urgent in needing attention to improve the security of the system under test.

Target Information

This lists what was tested and the type of testing carried out.

Methodology

This is a short, technical description, of physical way that the testing was undertaken.

Issue Summary

This is a high level, technical, summary of each issue found. It's common for this section to contain multiple examples of the same type of issue as we list every instance of an issue we find so that there is a clear list of everything that needs to be fixed – i.e. If we find a form that is insecure it's possible that each individual field within the form has the same type of issue and so if the form has five fields; First name, Last name, Email address, Company name and comments these would be listed as 5 issues as each field needs attention.

Vulnerability by Severity

This is a visual representation of the different levels of issues found to provide a quick overview of the number of each type individually and in relation to the application as a whole.

Details of Individual Issues

This is a detailed list of each issue found containing, where possible:

- Affected assets – what in the system has the issues
- Severity – on a 5-point scale from 'Critical' to 'For information only'
- OWASP / CWE mapping – how the issue relates to security testing methodologies or databases
- Technical description – a detailed technical explanation of the issue
- Steps to reproduce – how to verify the issue exists and to check that it has been fixed
- Business impact – a non-technical description of the potential impact of the issue
- Solution and/or mitigation – what's required to fix the issue
- External technical reference links – additional information in the public domain to provide more insight into the type of issue, other ways to fix it etc.

As with the issue summary, every instance of an issue is listed individually to provide a detailed, step-by-step guide for all fixes required.

Executive Summary

Appsecco was contracted by CLIENT NAME to conduct a Greybox Penetration Testing of SaaS based REDACTED Web Application and the SaaS based REDACTED Internal Network, defined in the target information section. The objective of this test was to determine if there were any security weaknesses in the SaaS based REDACTED Web Application or the hosts on the SaaS based REDACTED Internal Network that could render the application/infrastructure insecure and allow a malicious user to gain access to any data that is accessible via them or gain access to the underlying network/operating systems.

The assessment was carried out between START DATE and END DATE on the SaaS based REDACTED Web Application and the SaaS based REDACTED Internal Network.

The OWASP Top 10 2013, OWASP Top 10 2017 and CWE were used as the reference frameworks to evaluate and categorize any security issues discovered.

The Common Vulnerability Scoring System Version 3.1 calculator has been used to calculate the severity of any weaknesses that were discovered.

A time boxed, Greybox Penetration Testing was performed against the application in scope. During this assessment, we had access to following user accounts:

1. REDACTED
2. REDACTED

We had the ability to create additional users. New user accounts were created for following user roles:

1. REDACTED
2. REDACTED
3. REDACTED

We have created the following user accounts during testing:

1. REDACTED
2. REDACTED

A time boxed, Greybox Penetration Testing was performed against the SaaS based REDACTED Internal Network from the point of view of an attacker on the internal network. During the assessment. we had access to an internal machine on the network that we used as the initial foothold to perform the attacks.

Summary of Results

Multiple severity issues have been identified as part of the SaaS based REDACTED Web Application testing.

These vulnerabilities occur because of missing function level access control, Insecure Direct Object Reference (IDOR) due to which IP addresses that are not part of a user subscription can be fetched by tampering the REDACTED value in a vulnerable POST request made to the endpoint.

Although there are significant number of services visible to an attacker on the SaaS based REDACTED internal network. Most of the services are up to date and are implemented with secure defaults. The web applications hosted on the servers in the SaaS based REDACTED internal network are not vulnerable to low hanging web application related vulnerabilities.

A summary of the issues discovered in the SaaS based REDACTED Web Application is listed below:

- Least privileged REDACTED can view Personally Identifiable Information (PII) of higher privileged users by accessing “REDACTED”
- An REDACTED (with no edit permissions) can create dashboard templates
- An REDACTED (with no edit permissions) can modify contents of any dashboard
- Least privileged users, like REDACTED, can read action logs of higher privileged users
- Least privileged REDACTED can view personal details and log history of privileged users via IDOR in “REDACTED”
- It is possible to brute force a password for a known username using multiple credentials sets in the same request
- Extra parameters in server response reveal sensitive information
- Least privileged REDACTED can call “REDACTED” service to export any dashboard
- Least privileged REDACTED can export and import any dashboard via vulnerable REST endpoints
- IPs that are not part of subscription can be fetched via Insecure Direct Object Reference (IDOR) in “REDACTED”
- REDACTED REST framework version is revealed via OPTIONS HTTP request method

Conclusion

Multiple severity issues have been identified as part of the SaaS based REDACTED Web Application.

The environment was tested extensively for any authentication and authorization weaknesses. The testing showed that the application has insufficient authorization implemented for the low privileged (an REDACTED) user which allows a low privileged user to access sensitive information, perform delete/modify operations on resources that they are not authorised to modify/access.

Other than the low privileged (an REDACTED) user, the application is able to withstand attacks originating from an authenticated as well as an unauthenticated user session. This was tested by manipulating the tokens, cookies and HTTP methods etc.

The testing has showed that the application is built with security in mind against injection and reflection attacks that could lead to SQL/Command injections and HTML/JavaScript injection.

Even though the application implements rate limiting for login requests, we have noticed that a vulnerable endpoint REDACTED allows up to 20 login attempts in a single request making it easy for an unauthenticated user to guess the password using a dictionary based approach.

Although there are significant number of services visible to an attacker on the SaaS based REDACTED internal network. Most of the services are up to date and are implemented with secure defaults. The web applications hosted on the servers in the SaaS based REDACTED internal network are not vulnerable to low hanging web application related vulnerabilities.

The discovered issues need to be fixed as a priority as these can be abused to gain access to business sensitive information and any other data present in the application. Fixing these issues will give assurance to the users of this environment.

Greybox Penetration Testing

Target Information – SaaS based REDACTED Web Application

The purpose of this Greybox Penetration Testing was to determine if there were any security weaknesses in the SaaS based REDACTED Web Application that could render the application insecure and allow a malicious user to gain access to any data that is accessible via the application or gain access to the underlying network/operating systems.

The scope included following hosts and types of testing:

Application	Domain/IP Address	Platform	Technology	Type of Testing
REDACTED APP NAME	REDACTED URL	REDACTED	REDACTED	Web Application Security Assessment
REDACTED APP NAME	REDACTED URL	REDACTED	REDACTED	Web Application Security Assessment

Target Information – SaaS based REDACTED Internal Network

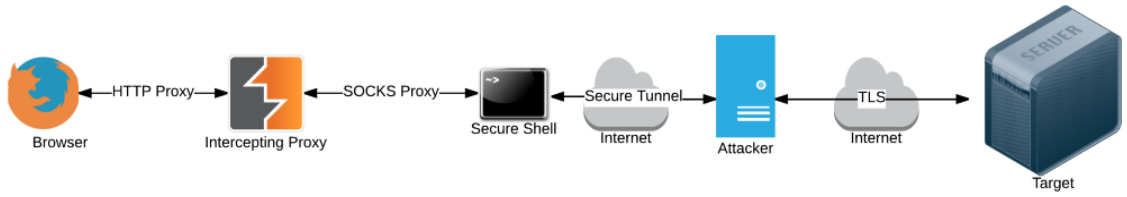
The purpose of this Greybox Penetration Testing was to determine if there were any security weaknesses in the services on the hosts in SaaS based REDACTED Internal Network that could render the infrastructure insecure and allow a malicious user to gain access to any data that is accessible via the application or gain access to the underlying network/operating systems.

The port scan results for SaaS based REDACTED Internal Network have been shared in a separate file named “REDACTED”.

Methodology

Testing Setup – SaaS based REDACTED Web Application

We setup an attacker machine so that all our attack traffic is from single source IP address. The attack traffic originated from the attacker IP - REDACTED. This IP was provided to CLIENT NAME before we began testing. The chain of traffic to the target was as below:

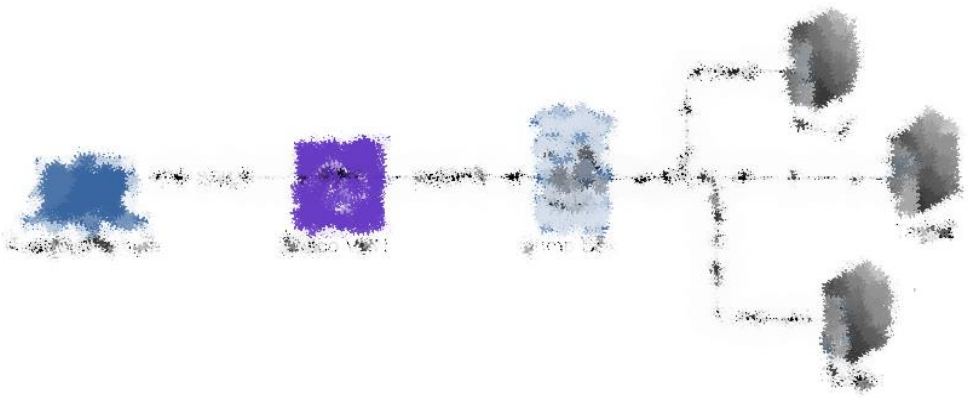


`Browser > Burp Suite Pro > Secure SOCKS Tunnel > REDACTED > TARGET`

Testing Setup – SaaS based REDACTED Internal Network

Before we began the testing, the following tasks were completed to setup the environment that would be used for the testing:

1. To gain access to the SaaS based REDACTED internal network, we used REDACTED for CLIENT NAME using endpoint REDACTED URL. The credentials for the REDACTED access were shared by CLIENT NAME with us prior to the assessment.
2. We were also provided SSH access to an internal machine on the network. The credentials for SSH were provided by CLIENT NAME to us prior to the assessment. We used this machine as initial foothold. All our attack traffic originated from this machine with the IP address – REDACTED



Approach

Greybox Penetration Testing - SaaS based REDACTED Internal Network

As an attacker, the following approach was taken while testing the SaaS based REDACTED Internal Network -

1. REDACTED

Issue Summary

1. Least privileged REDACTED can view Personally Identifiable Information (PII) of higher privileged users by accessing “REDACTED”

An REDACTED user, who has no privileges and is allowed to view only their own user details, can make a POST request to a vulnerable endpoint to fetch complete user information of a privileged user. The leaked details include REDACTED, sensitive info etc.

2. An REDACTED (with no edit permissions) can create dashboard templates

An REDACTED who does not have any permissions, except access to REDACTED and REDACTED modules, can create new dashboard templates

3. An REDACTED (with no edit permissions) can modify contents of any dashboard

An REDACTED who does not have any permissions, except access to REDACTED and REDACTED modules, can modify contents of any dashboard. By passing an empty REDACTED parameter, the dashboard contents could be cleared of all its data.

4. Least privileged users, like REDACTED, can read action logs of higher privileged users

A least privileged user (e.g., an REDACTED) can read action logs of higher privileged users (e.g. REDACTED).

5. Least privileged REDACTED can view personal details and log history of privileged users via IDOR in " REDACTED"

An REDACTED user, who has no privileges and is allowed to view only their own user details, can call REDACTED service to fetch user details of a privileged user. The leaked details include REDACTED, sensitive info, etc. This service can also be abused to read log history of privileged users.

6. It is possible to brute force a password for a known username using multiple credentials sets in the same request

Even though rate limiting exists, in general, the discovered vulnerable endpoint allows unauthenticated users to guess account passwords by allowing upto 20 login attempts in a single request.

7. Extra parameters in server response reveal sensitive information

Extra parameters are being returned in server response for several requests. Some of these parameters are sensitive in nature. E.g., login username of the REDACTED is revealed to an REDACTED.

8. Least privileged REDACTED can call " REDACTED" service to export any dashboard

Least privileged REDACTED can perform unauthorised operations by making direct call to methods of REDACTED service to download Dashboards

9. Least privileged REDACTED can export and import any dashboard via vulnerable REST endpoints

Least privileged REDACTED can export and import any dashboard by making direct server requests to vulnerable endpoints

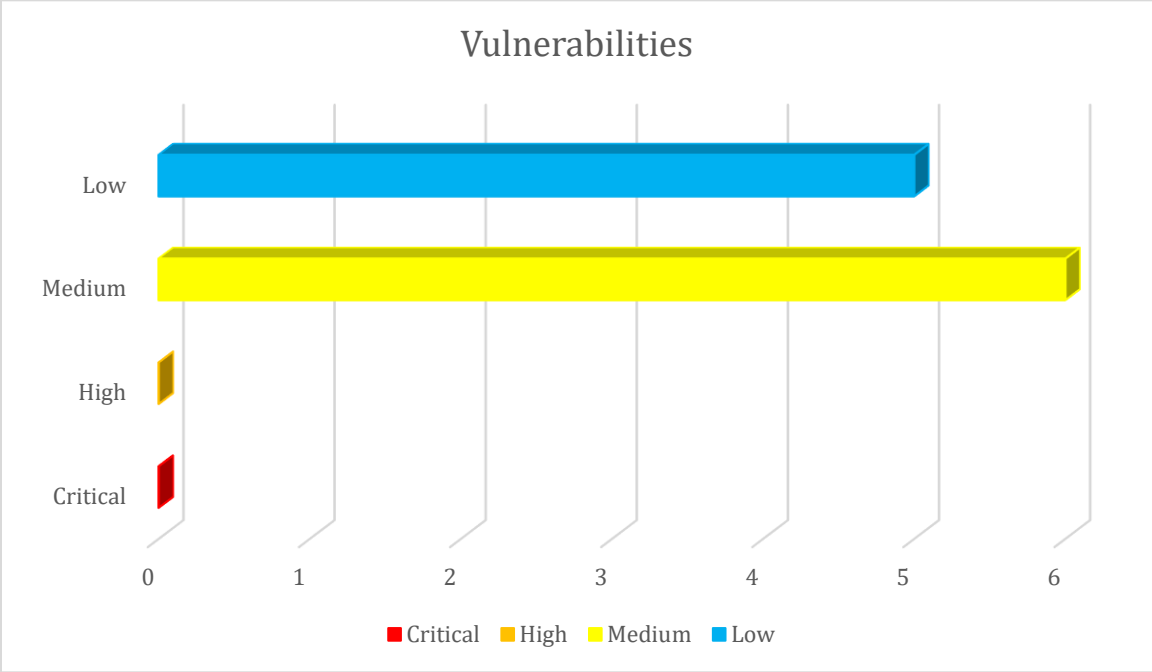
10. IPs that are not part of subscription can be fetched via Insecure Direct Object Reference (IDOR) in "REDACTED"

REDACTED and REDACTED can tamper the REDACTED in a vulnerable POST request, to fetch different sets of network IP addresses associated with arbitrary users and groups. This is applicable even if the REDACTED is not part of the same group or even if IPs are not part of the active REDACTED subscription.

11. REDACTED REST framework version is revealed via OPTIONS HTTP request method

REDACTED REST framework version details are obtained from server response body as REDACTED

Vulnerability Distribution by Severity



Summary of Issues by Severity

Severity	Count
Critical	0
High	0
Medium	6
Low	5

1. Least privileged REDACTED can view Personally Identifiable Information (PII) of higher privileged users by accessing “REDACTED”

Affected Assets

- REDACTED

Severity

Medium

OWASP / CWE Mapping

- [OWASP-2013-A7] Missing Function Level Access Control
- [OWASP-2017-A5] Broken Access Control

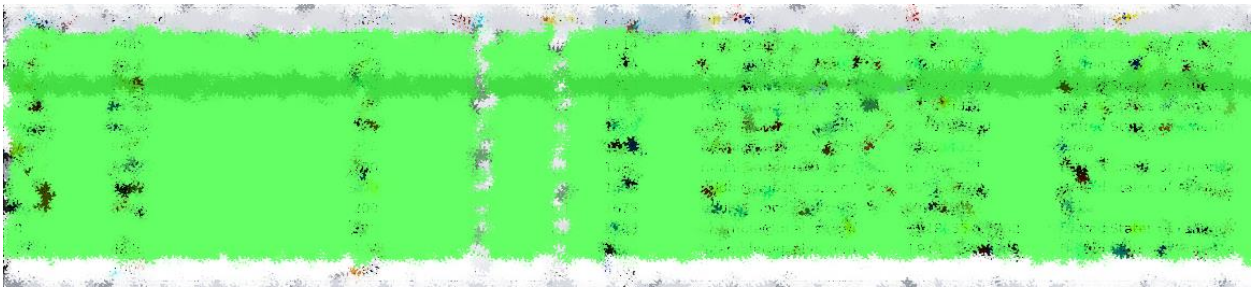
Technical Description

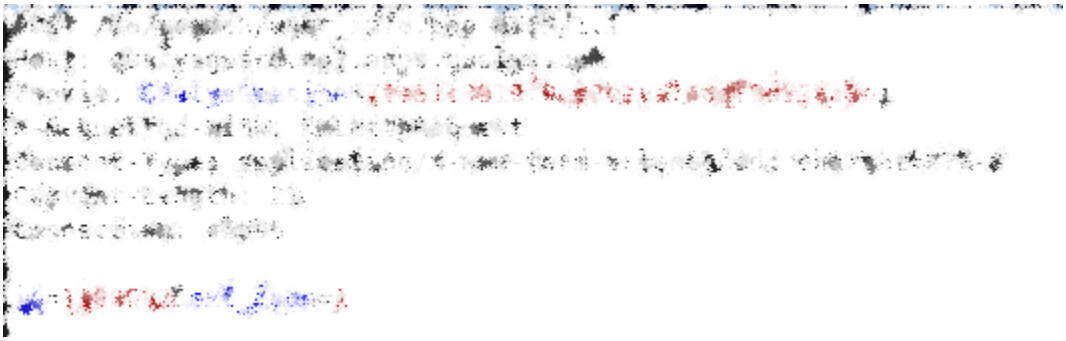
REDACTED

Steps To Reproduce

1. REDACTED

Screenshots





Business Impact - So What?

REDACTED

Solution

REDACTED

Mitigation

REDACTED

References

- REDACTED

2. An REDACTED (with no edit permissions) can create dashboard templates

Affected Assets

- REDACTED

Severity

Medium

OWASP / CWE Mapping

- [OWASP-2013-A7] Missing Function Level Access Control
- [OWASP-2017-A5] Broken Access Control

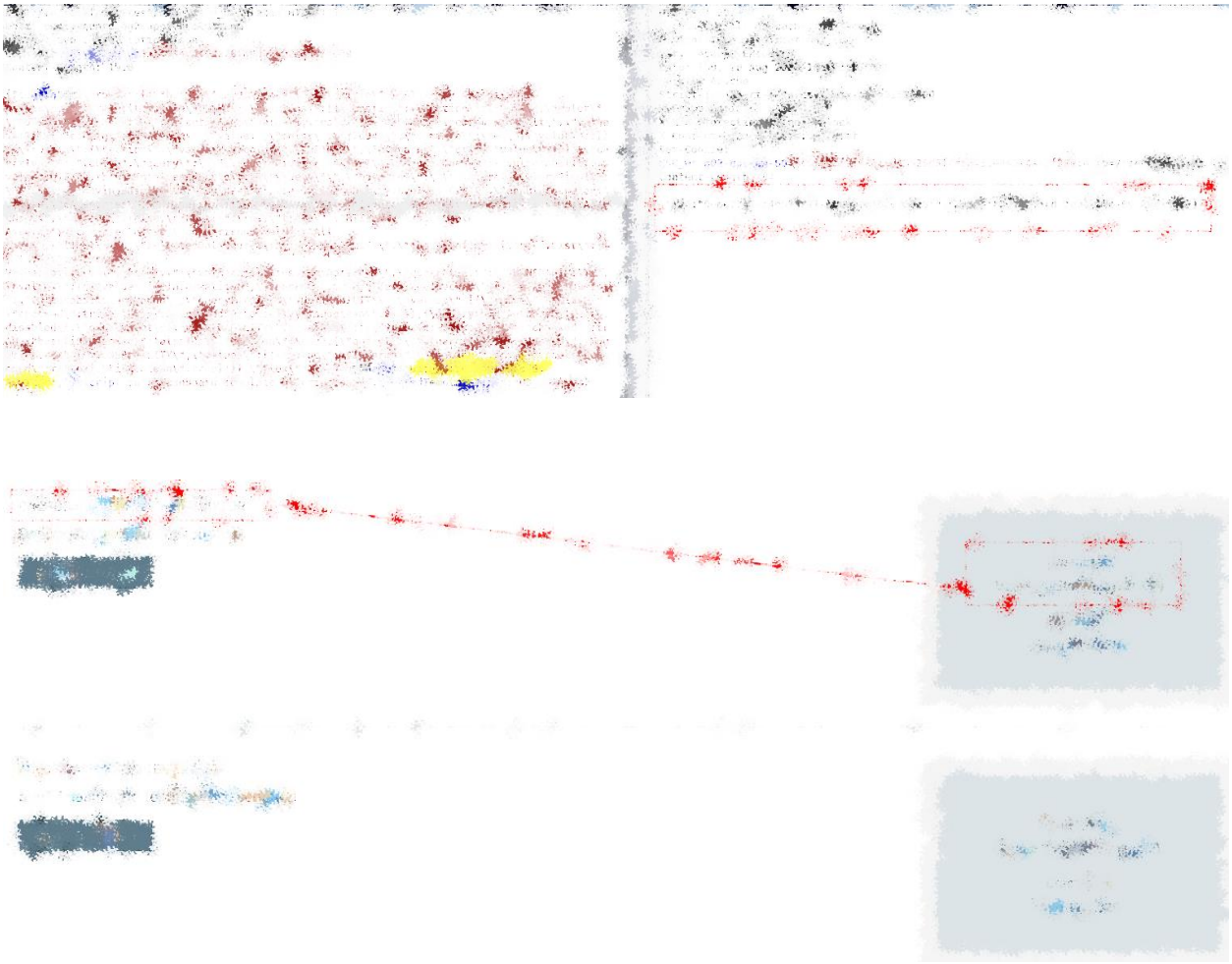
Technical Description

REDACTED

Steps To Reproduce

1. REDACTED

Screenshots



Business Impact - So What?

REDACTED

Solution

REDACTED

Mitigation

REDACTED

References

- REDACTED

3. An REDACTED (with no edit permissions) can modify contents of any dashboard

Affected Assets

- REDACTED

Severity

Medium

OWASP / CWE Mapping

- [OWASP-2013-A7] Missing Function Level Access Control
- [OWASP-2017-A5] Broken Access Control

Technical Description

REDACTED

Steps To Reproduce

1. REDACTED

Screenshots

Business Impact - So What?

REDACTED

Solution

REDACTED

Mitigation

REDACTED

References

- REDACTED

4. Least privileged users, like REDACTED, can read action logs of higher privileged users

Affected Assets

- REDACTED

Severity

Medium

OWASP / CWE Mapping

- [OWASP-2013-A7] Missing Function Level Access Control
- [OWASP-2017-A5] Broken Access Control

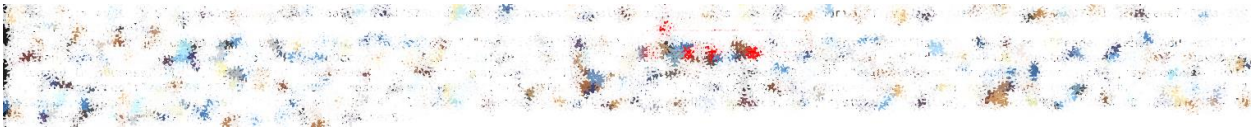
Technical Description

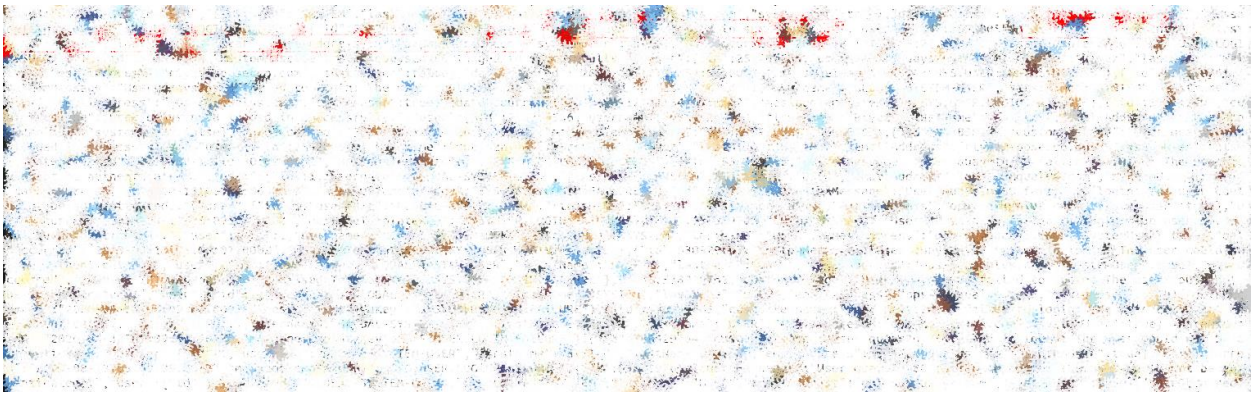
REDACTED

Steps To Reproduce

1. REDACTED

Screenshots





Business Impact - So What?

REDACTED

Solution

REDACTED

Mitigation

REDACTED

References

- REDACTED

5. Least privileged REDACTED can view personal details and log history of privileged users via IDOR in " REDACTED"

Affected Assets

- REDACTED

Severity

Medium

OWASP / CWE Mapping

- [OWASP-2013-A7] Missing Function Level Access Control
- [OWASP-2017-A5] Broken Access Control

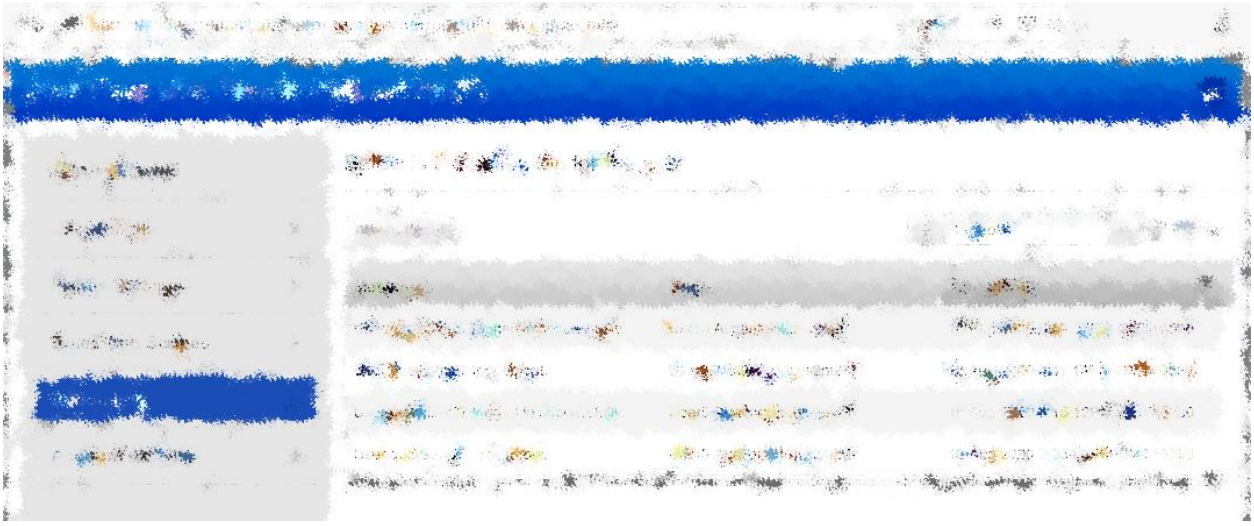
Technical Description

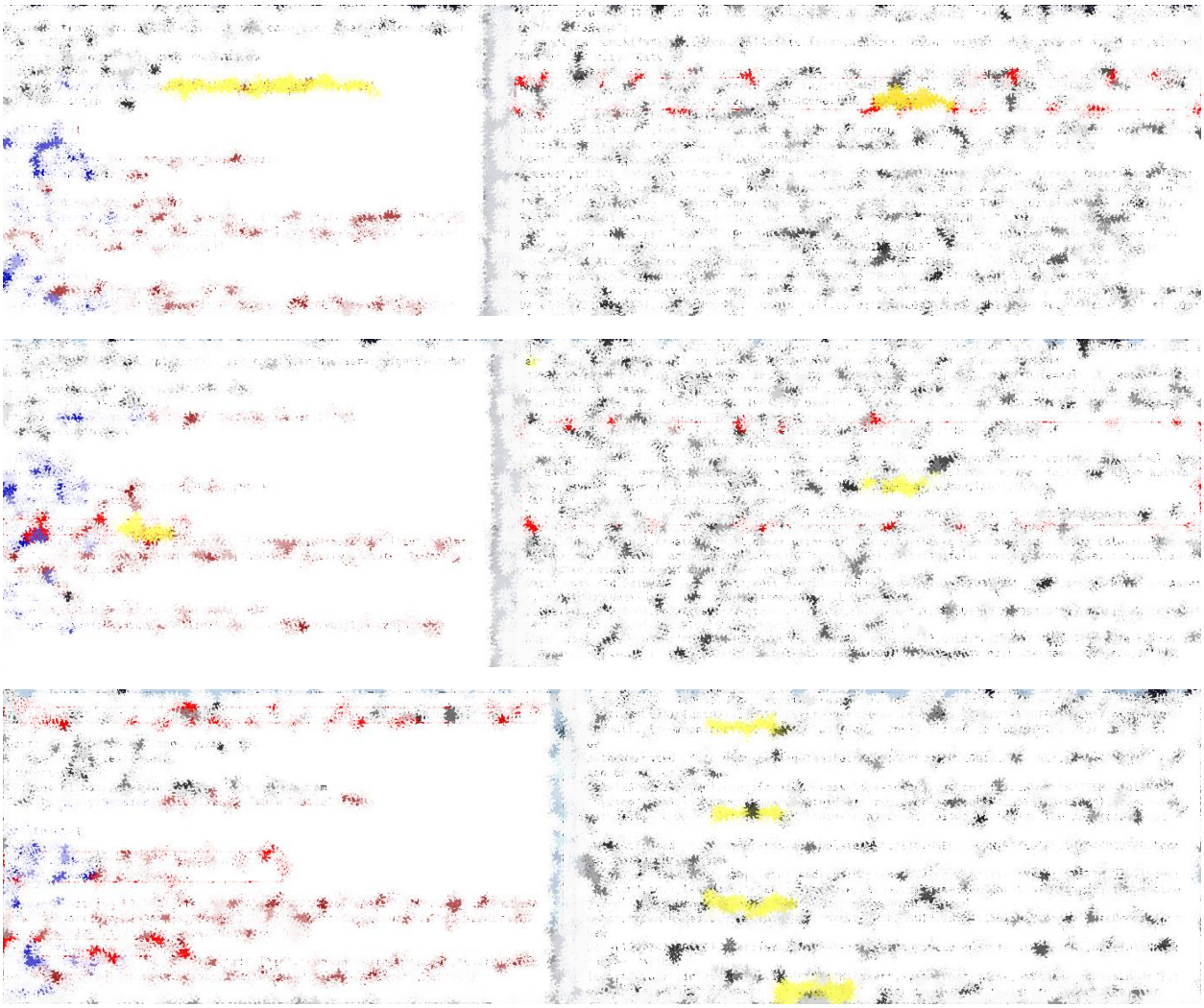
REDACTED

Steps To Reproduce

1. REDACTED

Screenshots





Business Impact - So What?

REDACTED

Solution

REDACTED

Mitigation

REDACTED

References

- REDACTED

6. It is possible to brute force a password for a known username using multiple credentials sets in the same request

Affected Assets

- REDACTED

Severity

Medium

OWASP / CWE Mapping

- [OWASP-2013-A5] Security Misconfiguration
- [OWASP-2017-A6] Security Misconfiguration

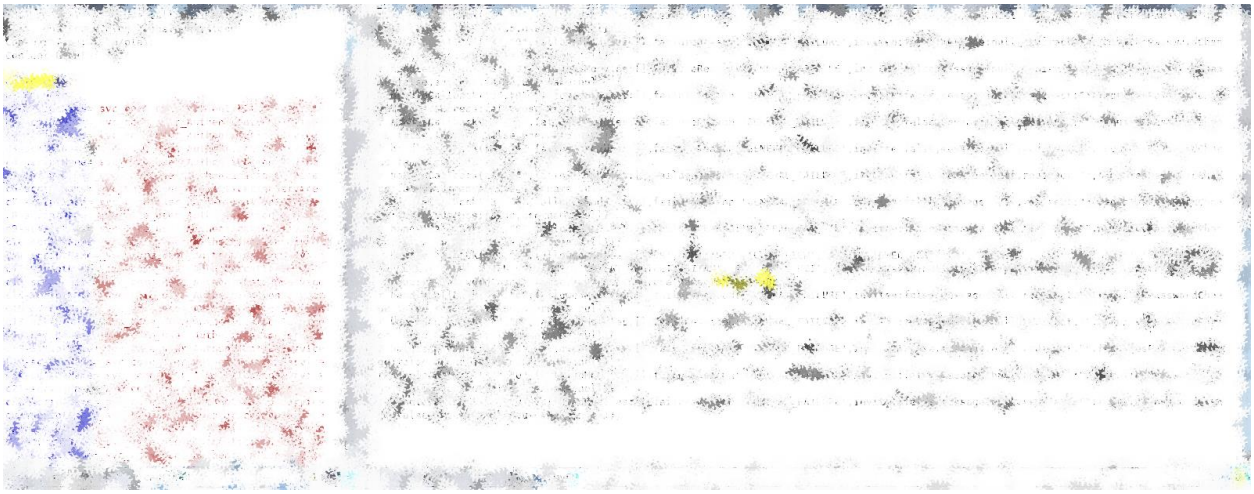
Technical Description

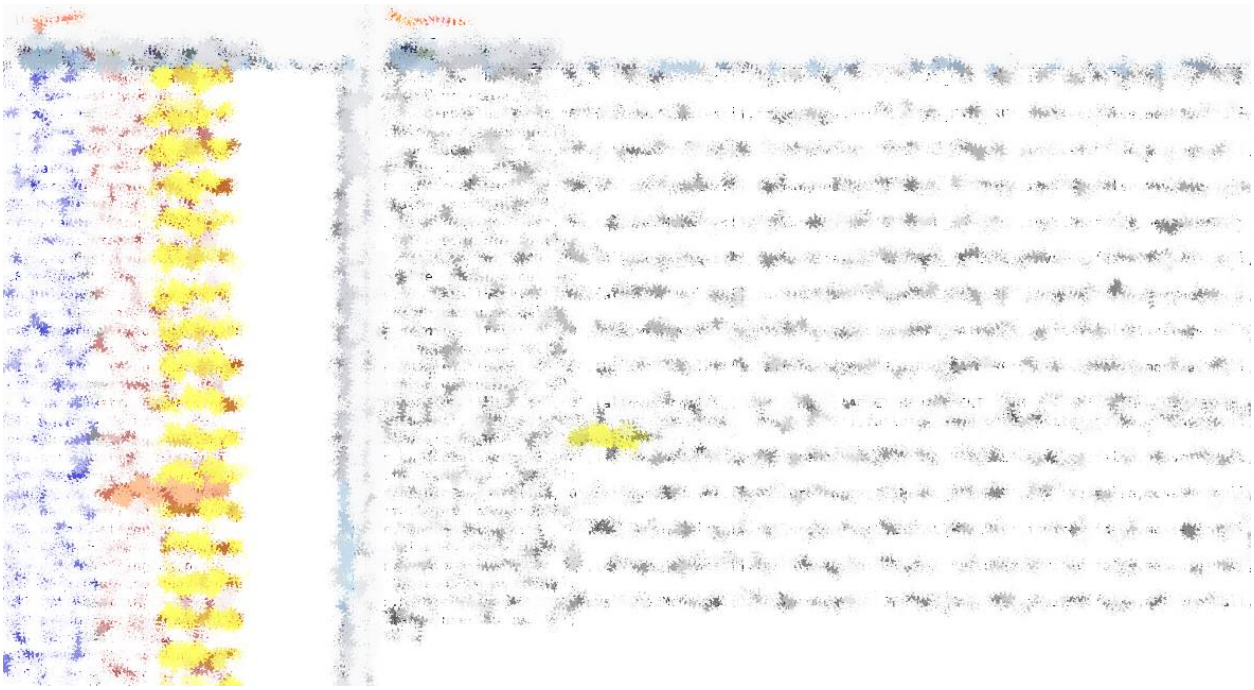
REDACTED

Steps To Reproduce

1. REDACTED

Screenshots





Business Impact - So What?

REDACTED

Solution

REDACTED

Mitigation

REDACTED

7. Extra parameters in server response reveal sensitive information

Affected Assets

- REDACTED

Severity

Low

OWASP / CWE Mapping

- [OWASP-2013-A5] Security Misconfiguration
- [OWASP-2017-A6] Security Misconfiguration

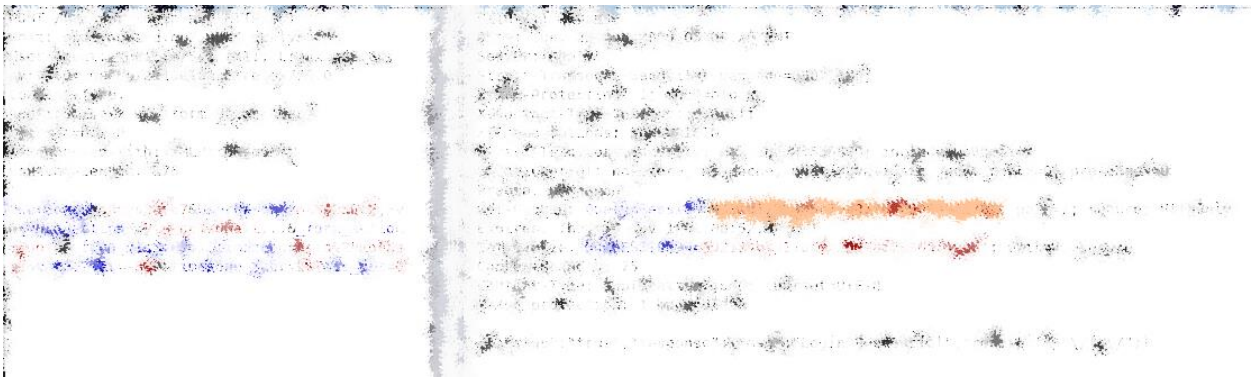
Technical Description

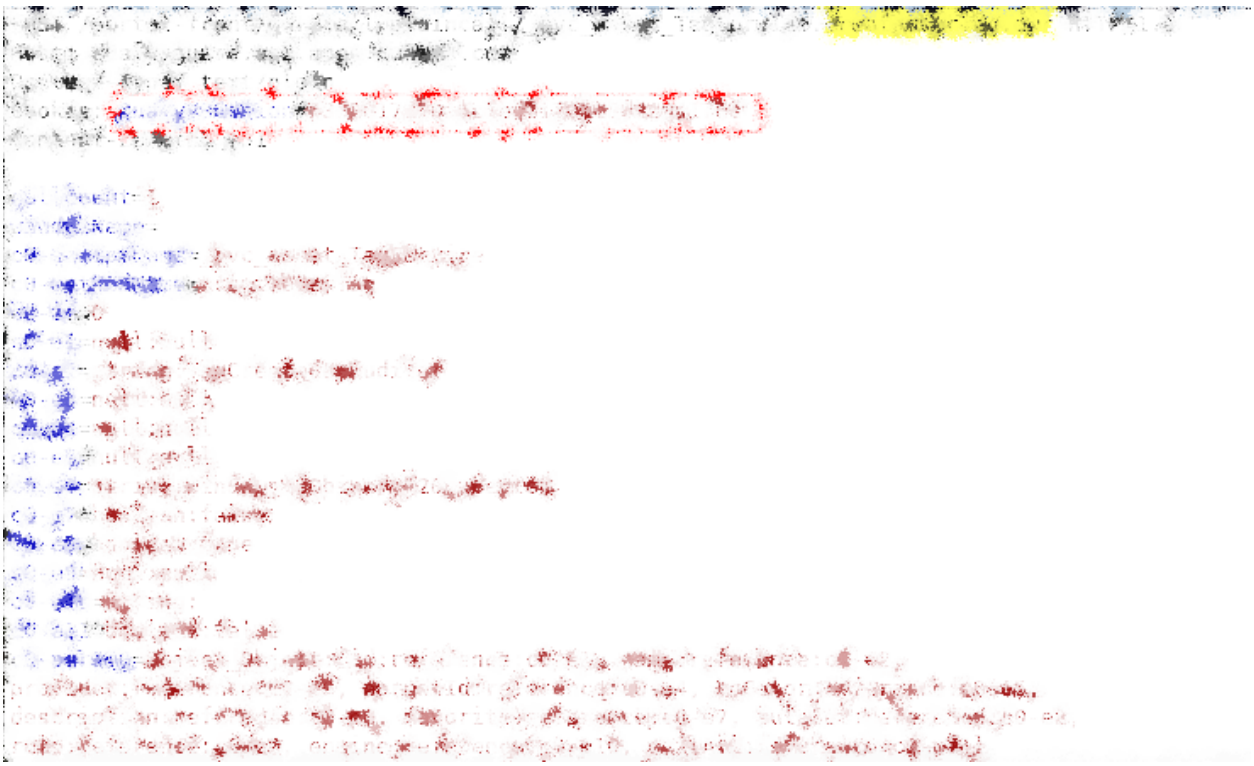
REDACTED

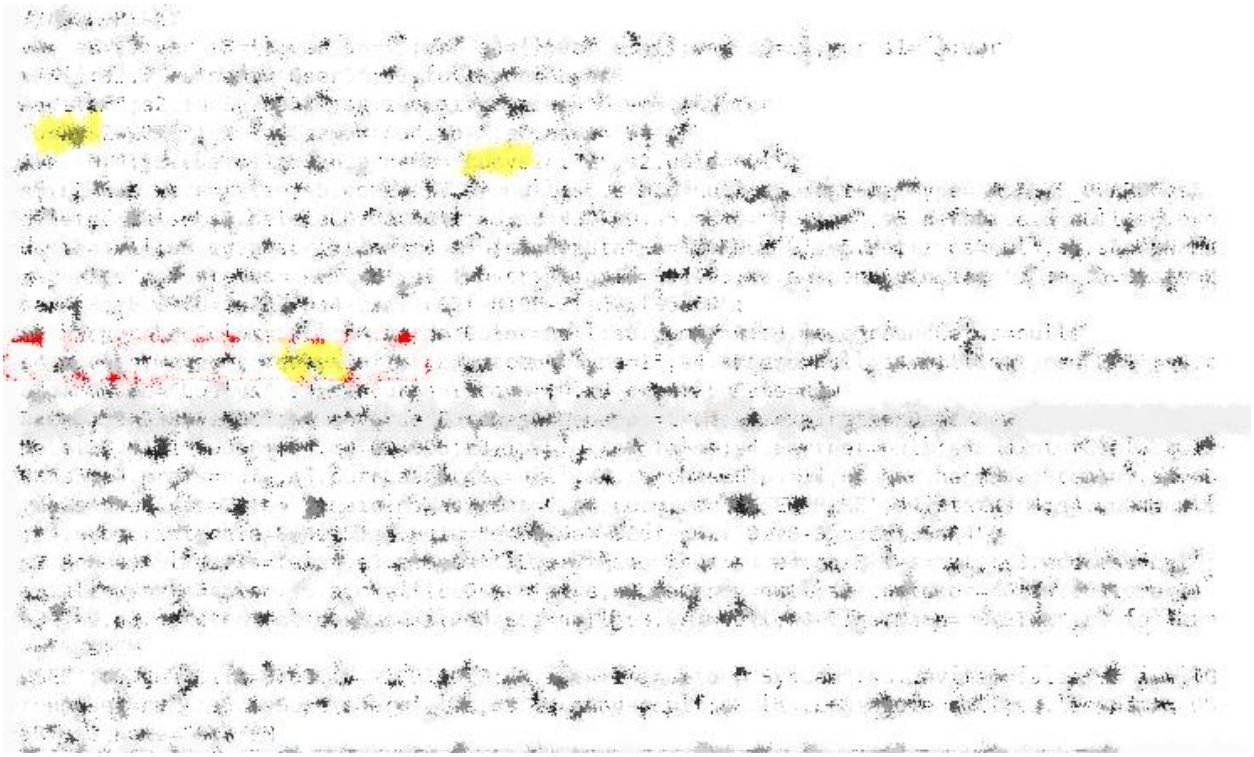
Steps To Reproduce

1. REDACTED

Screenshots







Business Impact - So What?

REDACTED

Solution

REDACTED

Mitigation

REDACTED

References

- REDACTED

8. Least privileged REDACTED can call "REDACTED" service to export any dashboard

Affected Assets

- REDACTED

Severity

Low

OWASP / CWE Mapping

- [OWASP-2013-A7] Missing Function Level Access Control
- [OWASP-2017-A5] Broken Access Control

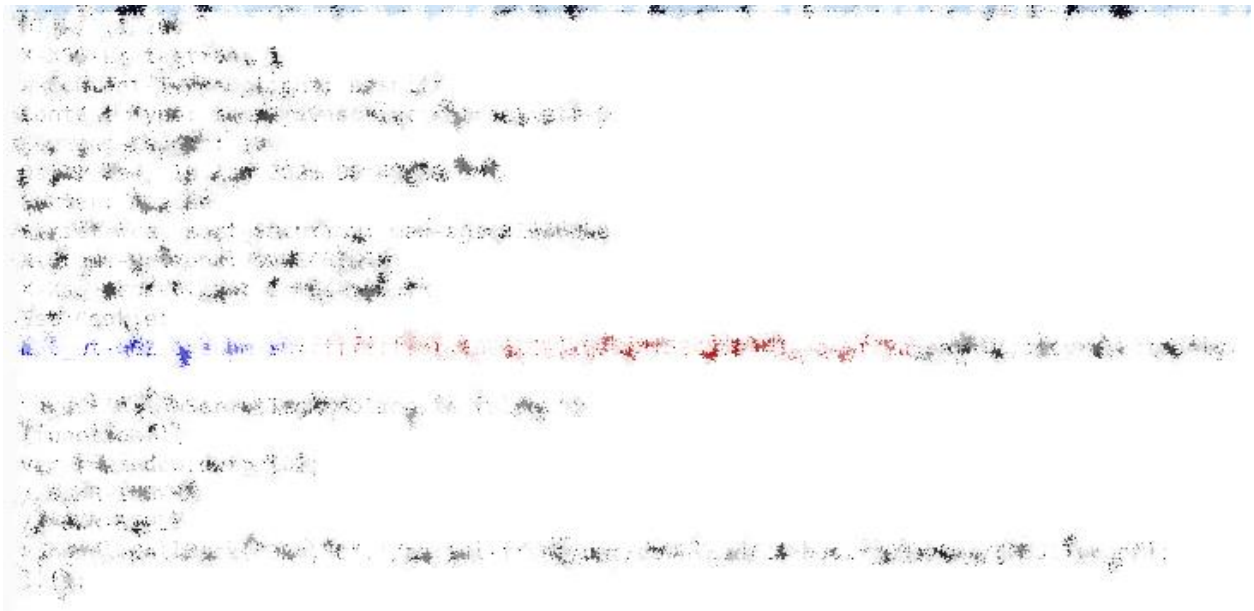
Technical Description

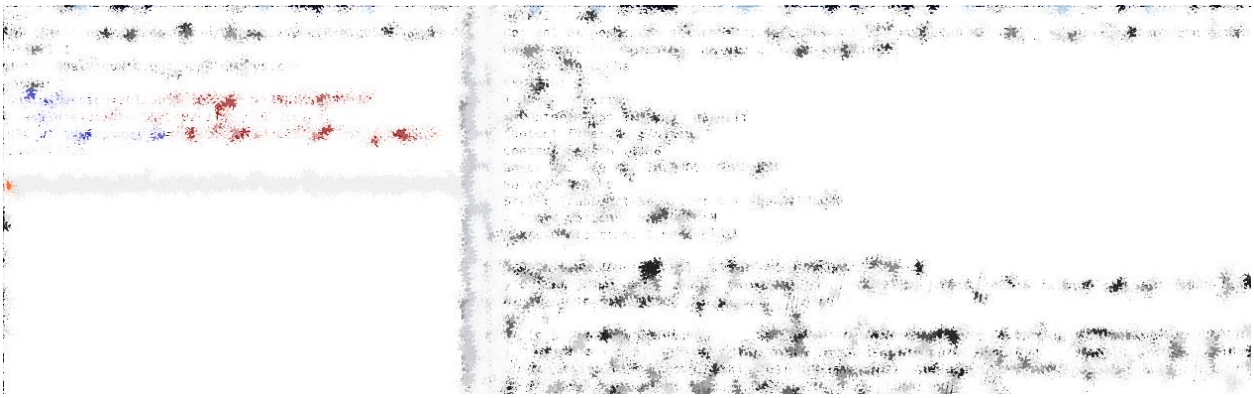
REDACTED

Steps To Reproduce

2. REDACTED

Screenshots





Business Impact - So What?

REDACTED.

Solution

REDACTED

Mitigation

REDACTED

References

- REDACTED

9. Least privileged REDACTED can export and import any dashboard via vulnerable REST endpoints

Affected Assets

- REDACTED

Severity

Low

OWASP / CWE Mapping

- [OWASP-2013-A7] Missing Function Level Access Control
- [OWASP-2017-A5] Broken Access Control

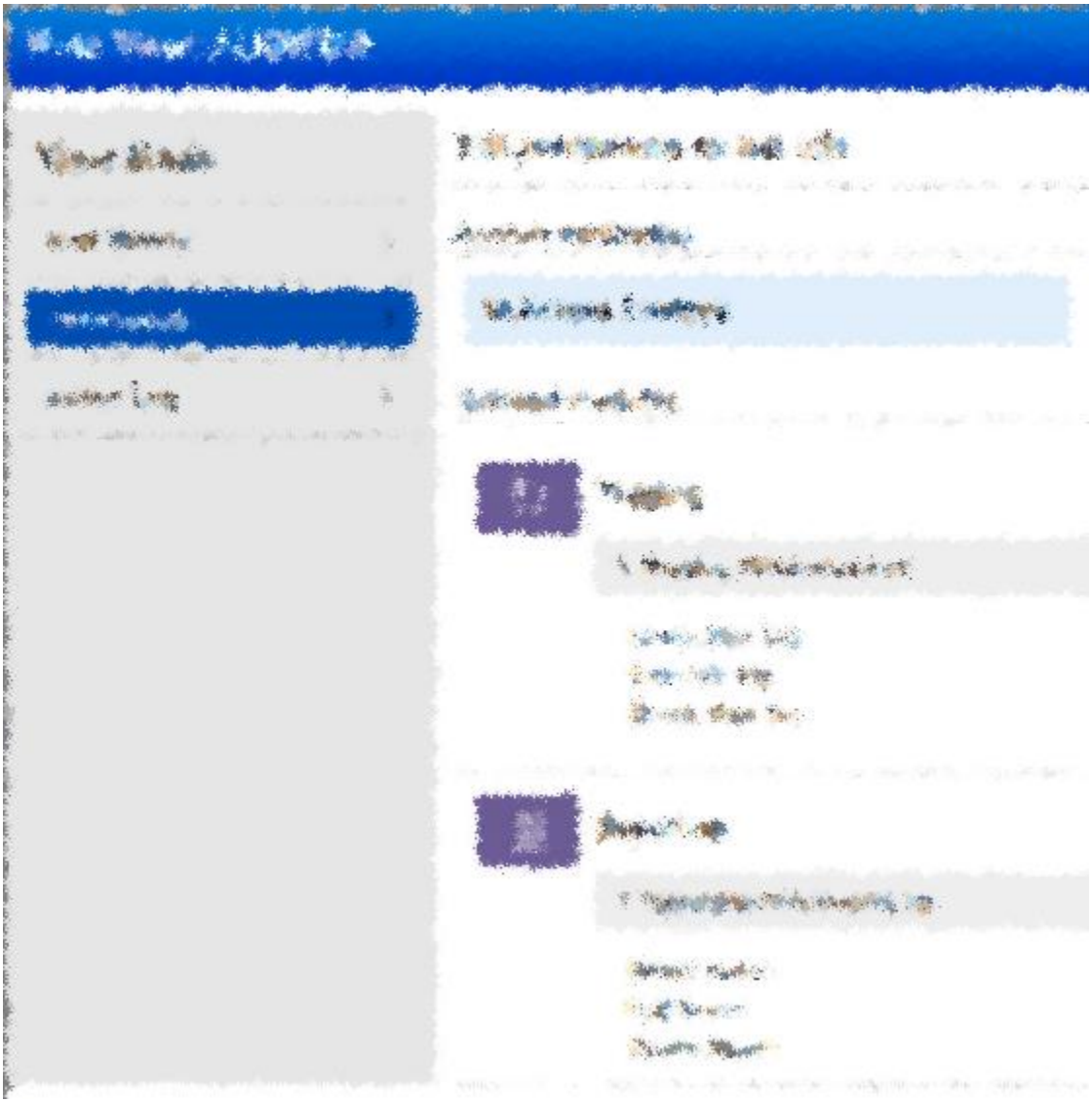
Technical Description

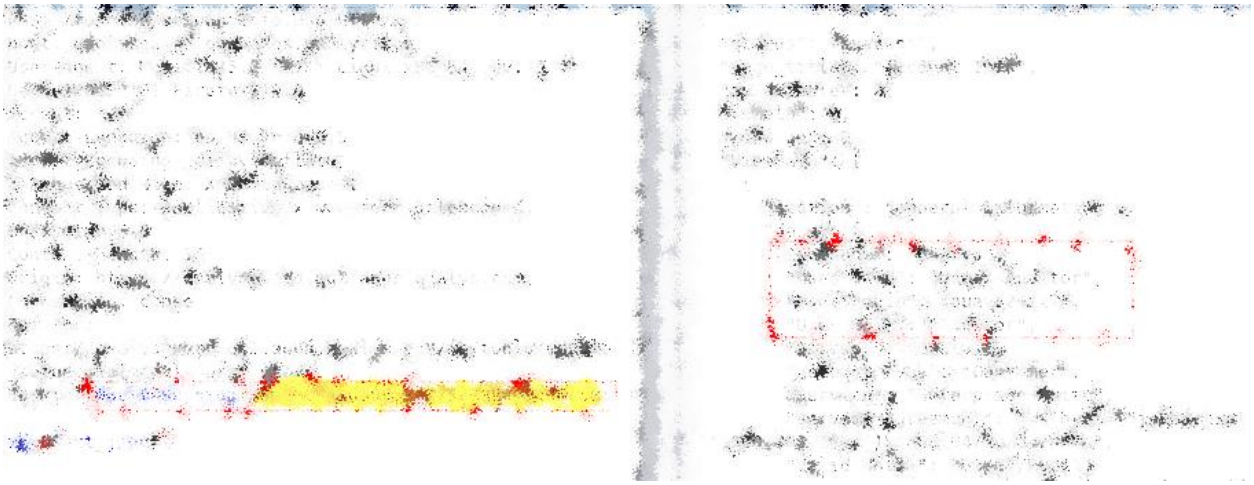
REDACTED

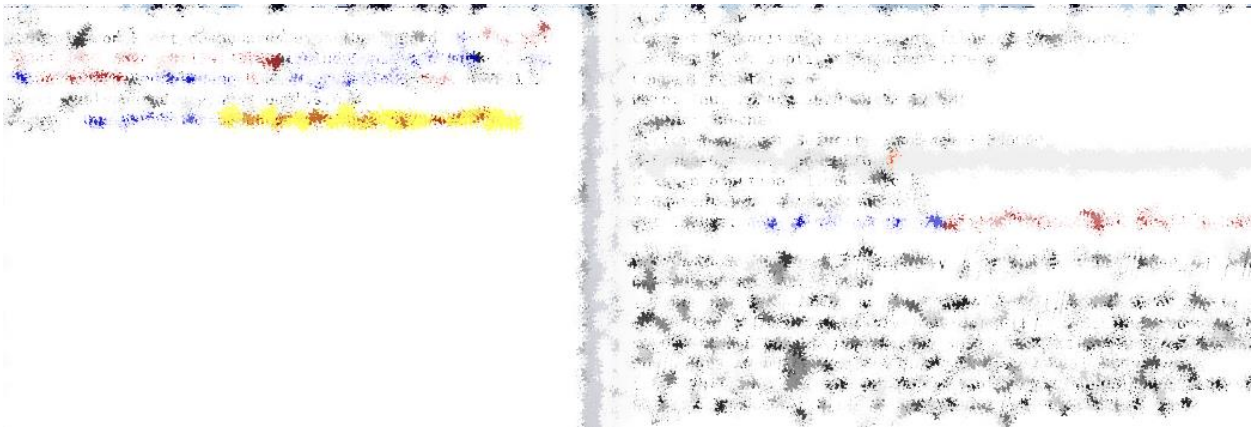
Steps To Reproduce

1. REDACTED

Screenshots







Business Impact - So What?

REDACTED

Solution

REDACTED

Mitigation

REDACTED

References

- REDACTED

10. IPs that are not part of subscription can be fetched via Insecure Direct Object Reference (IDOR) in “REDACTED”

Affected Assets

- REDACTED

Severity

Low

OWASP / CWE Mapping

- [OWASP-2013-A4] Insecure Direct Object Reference
- [OWASP-2017-A5] Broken Access Control

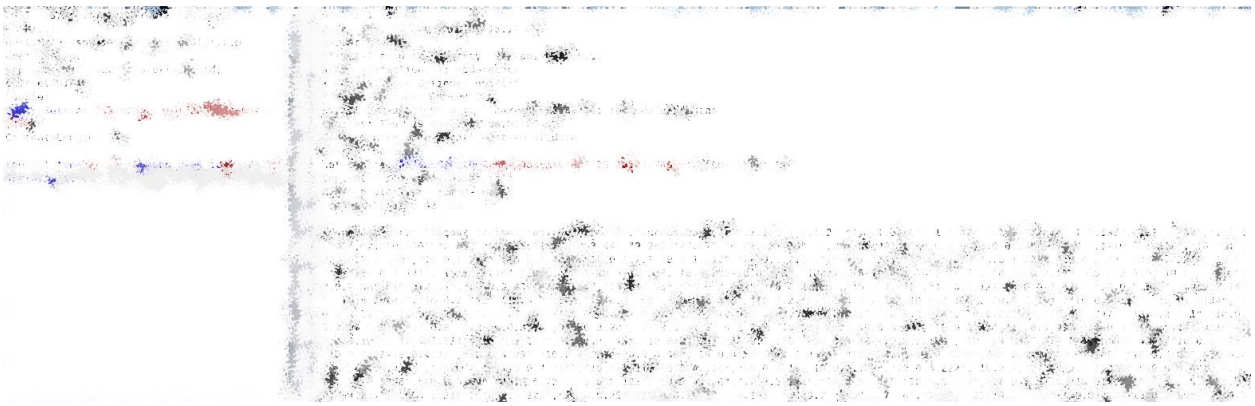
Technical Description

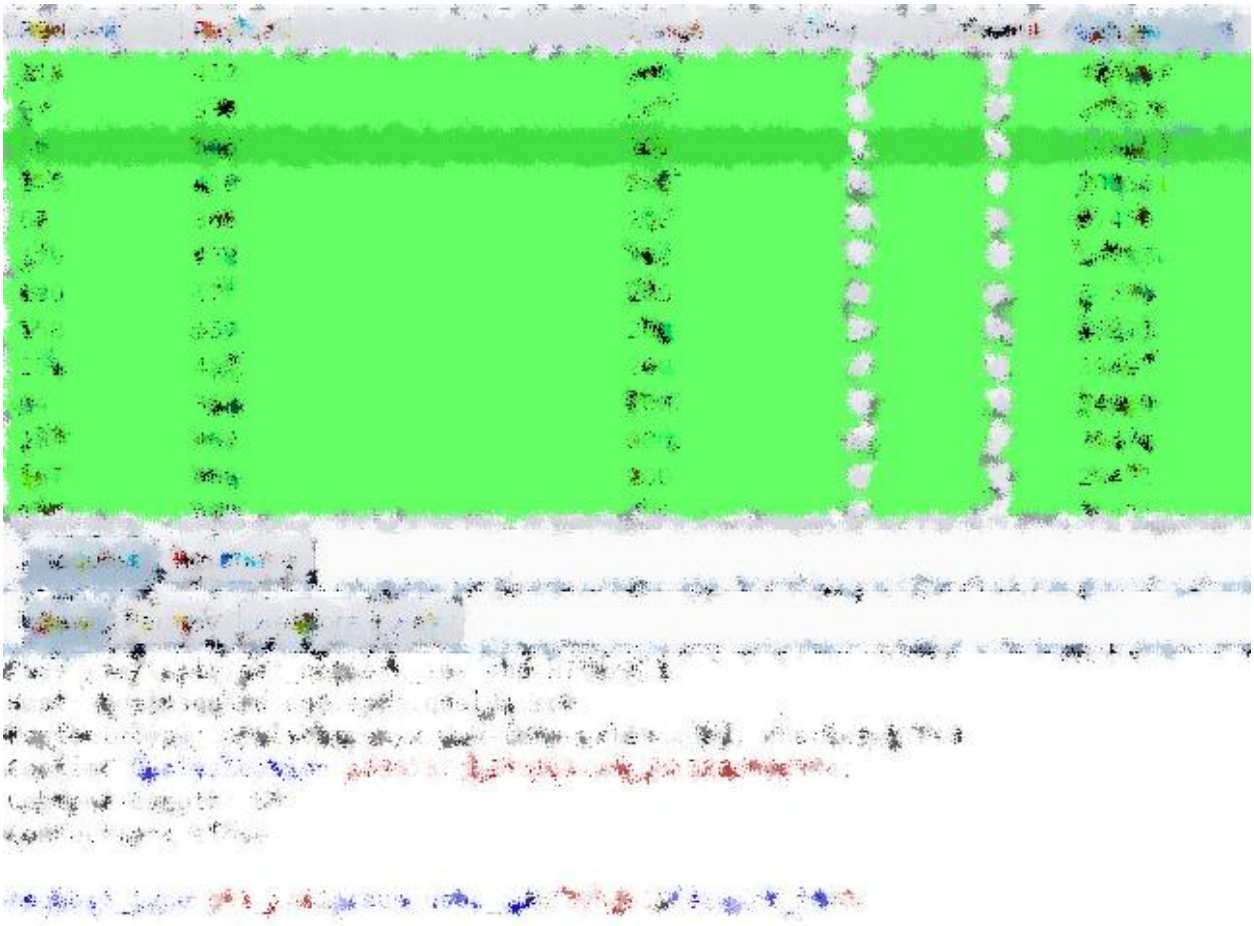
REDACTED

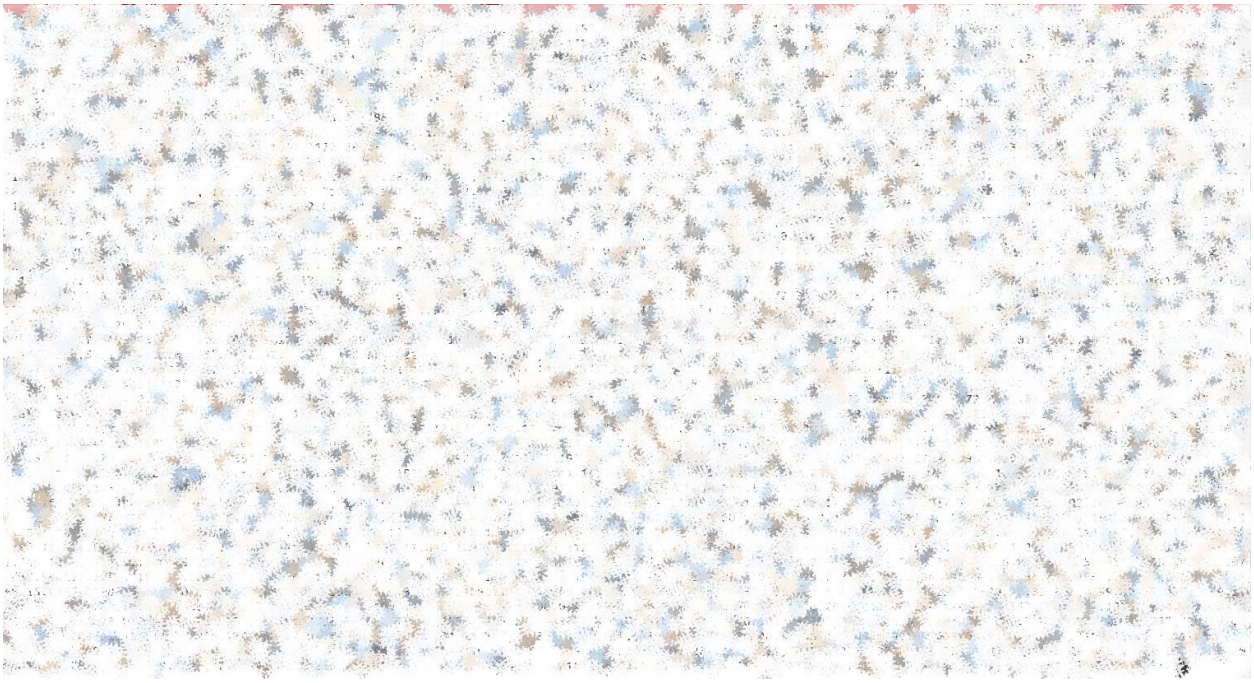
Steps To Reproduce

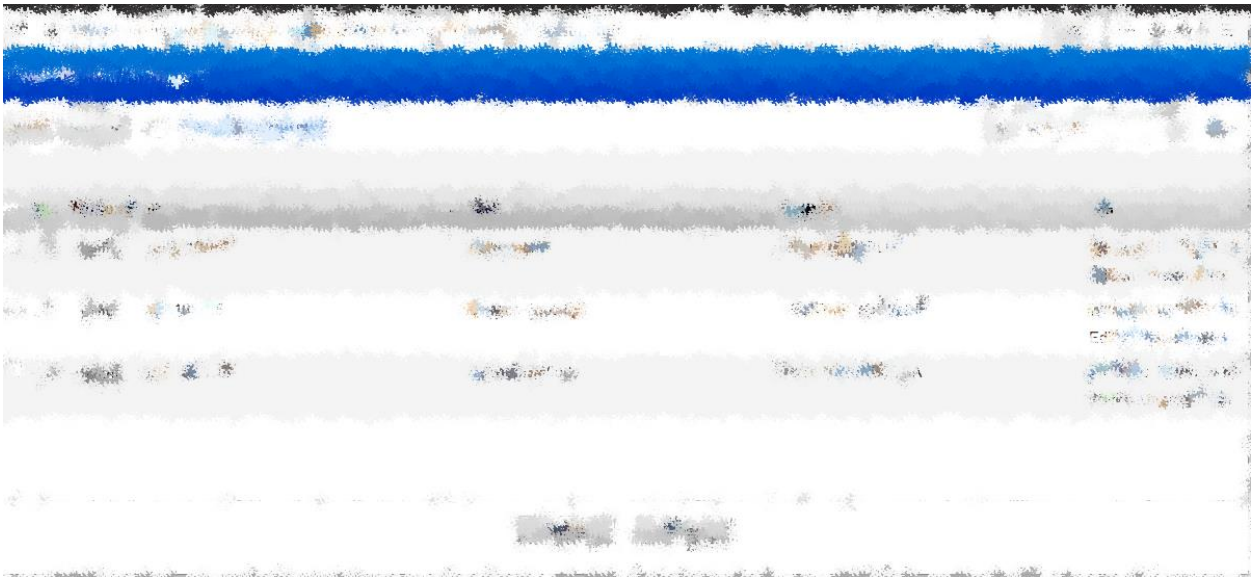
1. REDACTED

Screenshots









Business Impact - So What?

REDACTED

Solution

REDACTED

Mitigation

REDACTED

Recommendation

REDACTED

References

- REDACTED

11. REDACTED REST framework version is revealed via OPTIONS HTTP request method

Affected Assets

- REDACTED

Severity

Low

OWASP / CWE Mapping

- [OWASP-2013-A9] Using Components with Known Vulnerability
- [OWASP-2017-A9] Using Components with Known Vulnerabilities

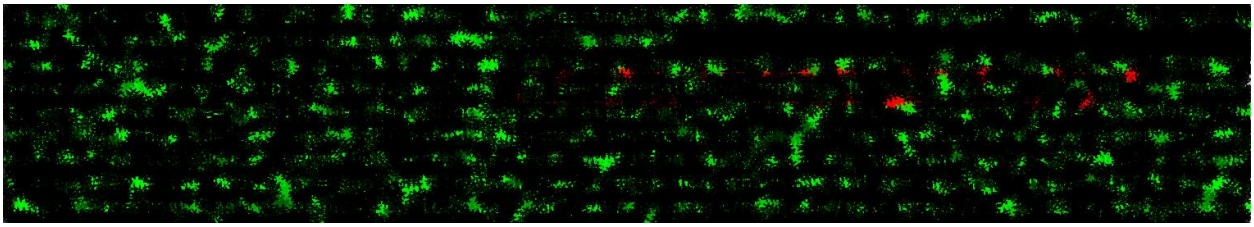
Technical Description

REDACTED

Steps To Reproduce

1. REDACTED

Screenshots



Business Impact - So What?

REDACTED

Solution

REDACTED

Recommendation

REDACTED

References

- REDACTED

Conclusion

Multiple severity issues have been identified as part of the SaaS based REDACTED Web Application.

The environment was tested extensively for any authentication and authorization weaknesses. The testing showed that the application has insufficient authorization implemented for the low privileged (an REDACTED) user which allows a low privileged user to access sensitive information, perform delete/modify operations on resources that they are not authorised to modify/access.

Other than the low privileged (an REDACTED) user, the application is able to withstand attacks originating from an authenticated as well as an unauthenticated user session. This was tested by manipulating the tokens, cookies and HTTP methods etc.

The testing has showed that the application is built with security in mind against injection and reflection attacks that could lead to SQL/Command injections and HTML/JavaScript injection.

Even though the application implements rate limiting for login requests, we have noticed that a vulnerable endpoint REDACTED allows up to 20 login attempts in a single request making it easy for an unauthenticated user to guess the password using a dictionary based approach.

The discovered issues need to be fixed as a priority as these can be abused to gain access to business sensitive information and any other data present in the application. Fixing these issues will give assurance to the users of this environment.