intruder

Web Application Penetration Test

Democrance

Tuesday, 10 September 2024

Contents

1. Executive Summary

Democrance Ltd. ("Democrance") approached Intruder Systems Ltd ("Intruder") with regards to performing a manual web application security assessment of their insurance and claims application, along with its supporting infrastructure.

The objective of the assessment was to discover security weaknesses in the application and infrastructure in scope, and to aid Democrance in securing their information systems from attack.

After the initial engagement, a re-test was performed to verify which of the issues previously found had been remediated. This report details security issues which were present after re-testing the issues found in the initial engagement.

Intruder assessed the security of the application to be very good. Of the security issues discovered through testing, two low severity issues and the defence-in-depth recommendations remained at the time of the re-test.

A summary of the low severity issues remaining is given below.

Low Severity Issues
The application exposes a static file that contains a list of subdomains and names of clients. This list includes domains of other client portals including domains not under Democrance's control. An attacker could use this list to perform further attacks on the exposed targets, or enumerate a list of Democrance clients.

The application uses a session cookie which is missing some security protections, increasing its risk of exposure.

Intruder recommends that the issues outlined in this report are fixed within a timescale appropriate to the severity of each issue. Only low severity issues remain, which is recommended to remediate at the next good opportunity.

Where appropriate, fixes should be re-tested to ensure that they have been remediated successfully.

2. Overview

Democrance Ltd. ("Democrance") approached Intruder Systems Ltd ("Intruder") with regards to performing a manual web application assessment against their insurance and claims application.

The objective of the assessment was to discover security weaknesses in the applications and APIs in scope, and to aid Democrance in securing their information systems from attack.

After the initial engagement, a re-test was performed to verify which of the issues previously found had been remediated. This report details security issues which were present after re-testing the issues found in the initial engagement.

Services Provided

Listed below is an overview of each of the services provided by Intruder to Democrance as part of this penetration testing engagement.

Manual web application penetration test

Intruder's skilled security specialists performed a thorough manual security assessment of Democrance's insurance and claims application. This also included an assessment of all supporting infrastructure, supporting services and APIs where applicable.

Democrance's application was assessed from both authenticated and unauthenticated perspectives to discover security weaknesses in the application implementation that could be discovered by malicious user, or attacker without valid credentials.

Intruder performed reconnaissance on the application's supporting infrastructure to discover other services and software which are exposed. All discovered services were assessed for vulnerabilities, insecure configurations and other known weaknesses.

In addition to manual testing, the assessment was also assisted by automated tools. All automated tool output was reviewed manually to review results for false positives and items which required further investigation.

Intruder's web application testing methodology contains more detail on the types of testing which were performed as part of this assessment, which can be found in Appendix A.

Scope

The scope of this assessment included a manual penetration test of the Democrance's web application. The targets in scope were:

- Admin Portal – https://staging.democrance.com/dmcadmin/login
- Broker Portal – https://dmc-staging.democrance.com/broker/login
- End-user insurance forms – https://dmc-staging.democrance.com/
- Dashboard Application – https://client-service.democrance.com/login

Authentication credentials were provided for the following users:

- pentest1 (Admin/Broker)
- pentest2 (Admin/Broker)
- pentest_super (Superuser)
- pentest1 (Dashboard Access)

Testing was performed from the perspective of a malicious user of each type listed above, as well as testing from an unauthenticated perspective.

Out of Scope

Denial of Service attacks which attempt to overwhelm target systems with traffic were considered out of scope for this engagement. Attacks of this nature are dependent upon resources available to either attacker or target, and as such do not add value to a security assessment.

Social engineering style attacks are not considered in scope and were not attempted as part of this assessment.

Physical penetration testing techniques such as gaining physical access to premises or systems were not considered in scope and were not attempted.

Timing

The security testing phase was performed over a period of five days, between 22 July and 26 July 2024. The reporting phase was one day. The re-test was carried out on 10 September 2024.

Terms and Conditions

The purpose of this service is to provide a level of assurance of the security of the systems in scope. However, Intruder Systems Ltd cannot guarantee the security of these systems at any stage before, during or after the assessment.

Due to the illegality of some of the techniques used by attackers, Intruder Systems Ltd are not able, and do not attempt to reproduce all possible attacks. Intruder Systems Ltd also operate on a fixed cost and time basis within which they aim to discover as many vulnerabilities which exist on the systems in scope. Due to these limits, Intruder Systems Ltd may not discover every possible weakness in the system that could be discovered by a real attacker who may have fewer restrictions on their activities or other resources.

Security assessment activities require simulating the activities of a real attacker. Although these activities are managed by experienced professionals with the intention of avoiding damage to the target systems, due to the nature of computer systems, the result of such activities cannot always be predicted.

Whilst care will be taken to avoid any intentional harm or incorrect information, Intruder Systems Ltd will not, at any time, be held responsible for any damages, including but not limited to, the loss of any intellectual property, assets, income, reputation or time, as a result of this engagement, either direct or indirect.

Document Control

Version	Date	Status	Author
0.1	29 July 2024	Initial draft	Raul Cicos
0.2	29 July 2024	Peer review	Daniel Andrew
1.0	30 July 2024	Issue	Raul Cicos
1.1	10 September 2024	Issue (Re-test)	Raul Cicos

3. Summary

The following table summarises the weaknesses found during the vulnerability assessment. The weaknesses are ranked by severity, which is a combination of their impact (if successfully exploited) with the difficulty of achieving the exploit. The methodology used to assess these ratings is given in the following section.

Reference	Severity	Title
DMC-2024-001	**Low**	Disclosure of Technical Information
DMC-2024-002	**Low**	Session Management Weaknesses
DMC-2024-003	**Informational**	Defence-in-Depth Recommendations

Intruder Severity Rating Methodology

Intruder aims to provide a sensible and consistent view on which security weaknesses should receive the highest priority for remediation. This is achieved with the following severity rating methodology, which ranks each weakness in terms of its severity. The severity is calculated according to the estimated impact if the weakness is successfully exploited and combined with the difficulty of achieving a successful exploit.

Information classification

Before understanding the impact of a weakness, we first need to define the differences in types of information. The following table gives a guide to three different levels of information sensitivity.

Classification	Description
Confidential	Information which is of significant value to the business, or requires protection under the Data Protection Act 1998. Examples: Customer names and addresses, email addresses, bank details, business plans, intellectual property, HR records.
Internal	Information which is not of significant value or is not afforded legal protection, but should not be released under normal circumstances. Examples: Network diagrams, source code, internal emails, reports, policy and procedure documentation, intranet content, newsletters.
Public	Information which is intended for public disclosure. Examples: Job postings, marketing materials, published financial records.

Impact

The aim of the impact rating is to understand what would realistically be expected to happen if the weakness is successfully exploited by an attacker. The impact is based upon a generic information classification system, provided in a table below, which aims to distinguish between the different types of information that may be at risk.

Impact	Criteria
High	The weakness allows an attacker to gain unauthorised access to a host, system, application, or to view or modify its data, and the data at risk is classified as Confidential. OR The weakness allows an attacker to cause denial of service, and loss of the system or service availability would cause revenue loss for the client. Examples: SQL Injection vulnerability on a database holding credit card data. A denial of service attack on a Payment Gateway. An authorisation weakness on an application such as a customer being able to view another customer's data.
Medium	The weakness allows an attacker to cause denial of service, or gain unauthorised access to a host, system, application, or to view or modify its data BUT loss of the system or service would not cause revenue loss for the client, or the system only contains Internal or Public data. Examples: Discovered administrator password on a developer's workstation. Gained access to file share containing budgets and project plans. Identified a weak password policy on an application which stores network information. Denial of service weakness on the corporate website.
Low	The weakness does not allow an attacker to gain any unauthorised access, deny any services or modify any information, but allows an attacker to gain further information about their target that may assist in such an attack. Examples: Service banners revealing version information. Directory listings enabled on a web server.

Difficulty

The aim of the difficulty rating is to determine how likely it is that this weakness might be successfully exploited by an attacker. This is measured by difficulty rather than likelihood, as the assumption is that the client is constantly under attack, but the easiest vulnerabilities will get exploited first.

Difficulty	Criteria
Trivial	The attacker requires no authentication, or can use publicly available access, and all information required to execute the attack is publicly available. **AND** The weakness can be exploited directly from the Internet (or other public network) and does not rely on the attacker being in a particular logical or physical location. **AND** The attack is simple and reliable and uses publicly available tools and exploits, and is almost guaranteed to be exploitable, requiring only the expertise of an intelligent malicious individual. *Examples: Heartbleed on internet facing web servers. Known default or trivially guessed passwords on internet facing routers.*
Moderate	The attacker requires some level of authentication not available to the public, or the attacker is required to gain access to a reasonably secure physical or logical location in order to perform the attack. **OR** The attack would require some moderate investment in time to execute or to develop custom tools or exploits, for example requiring the resources of an experienced cyber-criminal, or relies on another known vulnerability in the system. *Examples: Must have gained access to the client's office, have the credentials of a client employee or contractor, or have compromised a system in the DMZ. Or, a password policy is 'weak' but no weak passwords were actually found. Or, requires a client employee to click a malicious link from an email while they are signed into a particular web application.*
Complex	The attacker is required to gain access to a very specific or secure physical or logical location in order to perform the attack or relies on another hypothetical vulnerability being successfully exploited either before or afterwards to achieve the stated impact. **OR** The attack is borderline theoretical and may only work in rare cases, or requires significant effort to execute, for example, requiring the resources of a nation state. *Examples: Must be on the same unencrypted coffee shop wireless network as one of the client's customers while they sign into their online account. Or, must have physical access to the client's datacentre.*

Severity Rating

Once each issue has an impact and difficulty rating assigned, the following matrix is used to calculate a severity rating. The combination of impact and difficulty gives a sensible estimate of which weaknesses should be prioritised for remediation.

	Trivial	Moderate	Complex
High	**Critical**	**High**	**Medium**
Medium	**High**	**Medium**	**Low**
Low	**Medium**	**Low**	**Low**

4. Assessment Results - Web Application Penetration Test

DMC-2024-001 Disclosure of Technical Information

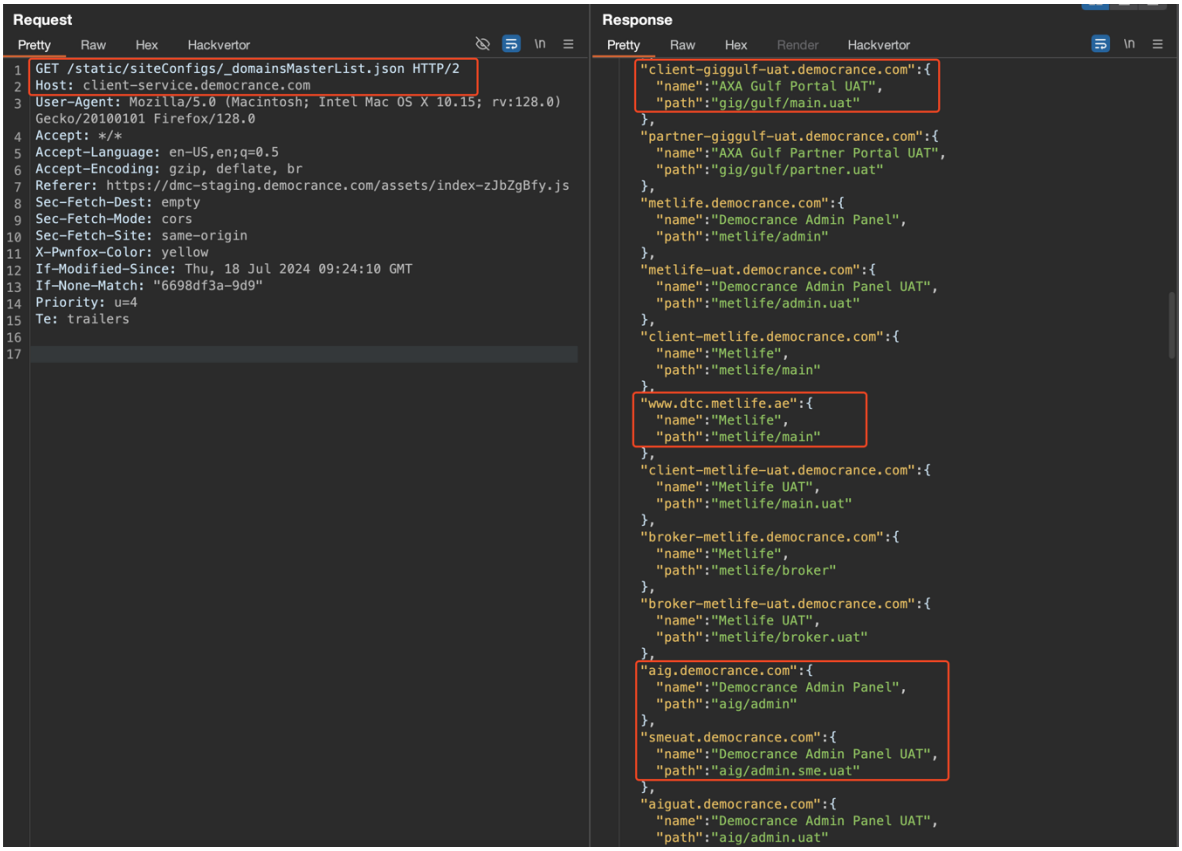
Severity:	Low	Impact:	Low
		Difficulty:	Moderate

Description

The application leaks technical data which could be useful to an attacker in mounting further attacks against the application and its users.

Evidence

It was possible to make an unauthenticated request to the server and receive a list of other client portals, client names as well as non-Democrance domains.



The technical information may be used by an attacker to help mount further attacks against the application and its users.

Remediation Advice

Disclosure of technical data can usually be prevented through proper server configuration and hardening. The request for “domainsMasterList.json” should be removed if not required by the application to function properly or it should be put behind authentication and/or limited to only required subdomains.

DMC-2024-002 Session Management Weaknesses

Severity:	**Low**	Impact:	**Medium**
		Difficulty:	**Complex**

Description

One weakness exists in the controls within the application designed to manage user authentication (session management controls). This weakness increases the likelihood of an attacker gaining unauthorised access to the application.

The application makes use of the following cookie to track user sessions:

```
session-store
```

The "session-store" cookie is not marked ‘HTTPOnly’. Marking a cookie ‘HTTPOnly’, prevents malicious JavaScript from reading the session token. This provides a rudimentary protection against some types of cross-site scripting attacks where attackers attempt to steal a legitimate user’s session and use it to impersonate them or perform malicious actions on their behalf.

Remediation Advice

When setting sensitive cookies in application responses, be sure to mark all sensitive cookies (such as session cookies, and cookies used to store CSRF tokens) as 'HTTPOnly'.

For more information on the 'HTTPOnly' cookie attribute, please see here.

DMC-2024-003 Defence-in-Depth Recommendations

Severity:	Informational	Impact:	N/A
		Difficulty:	N/A

Description

This section outlines additional 'defence-in-depth' measures which Intruder recommends to further improve the application's security. Defence-in-depth measures add layers of security controls which, when combined, provide redundancy in the event other security controls fail or vulnerabilities are exploited.

Remediation Advice

Each recommendation is outlined below, along with links to useful resources or further information.

No Multi-Factor Authentication

Add the option for application users to protect their account with a second factor of authentication. This helps prevent password spraying and credential stuffing attacks. More information can be found in this OWASP article.

No Content Security Policy

The website does not implement a Content Security Policy (CSP) across all responses as a secondary layer of defence against client-side attacks such as cross-site scripting. A strong CSP can prevent such attacks, or make them difficult to exploit. For more information, please see the following articles:

https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html
https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP

Staging Environment Exposed

The staging environment is publicly exposed to the internet and allows anyone to register as a customer and access to the administrative login panels. To further secure the staging environment against exploitation, consider moving it behind a VPN.

Concurrent Sessions

When possible, a user should only be allowed one active session at a time. When the user logs in for a second time, the previous session should be invalidated. Allowing multiple logins may encourage account sharing which reduces accountability and the user would have no indication if their account has been compromised.

Appendix A - Intruder Application Testing Methodology

The following table shows the Intruder application testing methodology at a high level. It gives an overview of all the categories of testing that we perform, along with examples of each of the types of tests that may be performed in each category during the assessments.

For further details of specific individual tests, please refer to the OWASP Application Security Testing Guidelines, which we align our testing to.

Category	Description
Authentication and session handling	Intruder look for weaknesses in the authentication and session handling mechanisms of the application, which could allow an attacker to: • bypass the authentication • authenticate as someone else • determine valid usernames or passwords for the application • manipulate the application to think the user is someone else • manipulate information the application believes about the user
Authorisation	Intruder look for weaknesses in the mechanisms of the application which enforce authorisation controls, which could allow an attacker to: • gain unauthorised access to data or privileges which are reserved for higher level users • gain unauthorised access to data or privileges which are confidential to other users or should otherwise be restricted
Business logic	Intruder look for weaknesses in the application mechanisms which are responsible for enforcing the business logic of the application, such as: • workflow or separation of duties functionality • pricing, subscription and licensing functionality • shopping cart, check-out, refunds and delivery processes
Input Validation and Code Injection	Intruder look for areas of the application where the user is able to introduce their own code and influence the operation of the application for themselves or other users. Some examples of these types of 'injection' attack are commonly known as: • HTML Injection (Cross-Site Scripting) • SQL Injection • XML/XPath Injection • Command Injection
File Uploads	File upload functionality is assessed for weaknesses which could allow an attacker to: • compromise the application and the server which supports it

	• use the server for file storage or distribution other than the purposes intended • support further attacks against the application or company
Information Leakage	Intruder assess the application looking for areas which disclose implementation details unnecessarily. For example: • what language, software, frameworks and their versions are in use • details about the operating system, its configuration or file system layout • details about other customers or users of the application
Encryption settings	Intruder assess encryption used in the application for robust configuration and implementation. For example, looking for weak encryption ciphers or key negotiation which could allow an attacker to: • gain access to read or modify the encrypted traffic of other users • bypass the integrity of messages believed to be secured against tampering with encryption
Server configuration	Intruder assess the server for configuration settings which may reduce the security of the server and the application. Some examples of these types of flaw are: • Missing HTTP security headers • Insecure CORS headers • Insecure direct object references • Cross-site request forgery (CSRF)

5. About Intruder

Intruder Systems Ltd is an independent security advisory company, specialising in providing continuous security monitoring for internet facing web applications and infrastructure.

Intruder consultants apply their industry-leading technical skills to deliver an approach which focusses on finding exploitable vulnerabilities with real business impact. This efficient testing and reporting style means more consultancy time is spent searching for serious issues, and less on reporting low impact and informational findings which could be discovered by automated scanners.

Intruder aims to deliver the highest calibre of security assessments in the industry, with a focus on technical excellence, risks presented in the context of realistic scenarios, and delivered with the utmost quality. For this reason, Intruder's penetration testers are accredited with professional qualifications from CREST, Offensive Security, and other equivalent leading bodies.

Intruder is SOC 2 type 2 accredited, and accredited with the Cyber Essentials scheme, under the IASME certifying body, to ensure high standards of security compliance.

Intruder is a member of the Cyber-security Information Sharing Partnership.

The Cyber-security Information Sharing Partnership (CiSP) is a joint industry government initiative to share cyber threat and vulnerability information in order to increase overall situational awareness of the cyber threat and therefore reduce the impact on UK business.

www.intruder.io contact@intruder.io @intruder_io

Intruder Systems Ltd, 71-75 Shelton Street, Covent Garden, London, England, WC2H 9JQ