Management response to Trustwave Penetration Test October 2021

To whom it may concern,

Lexer is pleased to provide you with its updated Penetration Test findings which were conducted by Trustwave in October 2021 ("**Test Report**").

Trustwave's Test Report found that Lexer's applications and underlying infrastructure adhered to "Best Practice" standards, and noted that: "*the Lexer applications and the underlying infrastructure [were] robust and resilient to external attacks.*"

We are pleased that the Test Report has not found **any** issue which would incur a consequence rating (Insignificant to Catastrophic, as described in clause 14 of the Test Report).

While no actionable vulnerabilities were found, Trustwave noted that the Lexer AWS environment could be improved upon (refer to section 3.1 of the Test Report, or AWS-1). Lexer welcomes this additional feedback on how it can continually improve the security of its infrastructure, and will consider the recommendations of Trustwave in its 2022 security roadmap.

Trustwave Finding			**Lexer Management Response**	
Risk	**Ref.**	**Weakness**	**Status**	**Comment**
Best Practice	AWS-1	Segmentation: Software Defined Customer Segmentation	No immediate action required - for information only	Lexer will consider Trustwave's recommendations as part of its 2022 security roadmap.

Noting that the designation "Best Practice" has not been defined in the Test Report, we have confirmed with Trustwave that, for the purposes of their report, it means:

> *For those observations/shortcomings where we cannot ascertain the impact or likelihood (potentially due to the lack of visibility of the attack surface), but at the same time there are security best practices available, we usually mark it as "best practice".*

If you have any questions please email us at security@lexer.io.

Regards,

The Lexer Information Security Team

Enc: Trustwave Test Report

Trustwave®

Project Ref. JOB21357

By Victor Kahan

Lexer - Annual Penetration Test Report

This document details the security posture of the Lexer External Facing Systems based on the findings identified by Trustwave during the External Penetration Test and AWS Infrastructure Security Review performed in July 2021. The report has been updated to reflect the status of remediation based on the remediation test performed in October 2021.

Summary

As part of Lexer on-going security assurance, Trustwave conducted a security assessment of the Lexer customer environment. The purpose of the security test was to assess the security posture of the Lexer application and underlying infrastructure against common external attacks.

Trustwave observed the Lexer applications and the underlying infrastructure robust and resilient to external attacks. In addition, Trustwave observed that the AWS environment relies on software-defined segmentation rules to segregate Lexer's customer environment and data.

Trustwave performed a remediation test on the 19th of October 2021. The report has been updated to reflect the status of the issue that was found to have been remediated.

Target Systems	Risk Level
• Lexer application and underlying infrastructure	**Best Practice**

All the previously reported security vulnerabilities were remediated by Lexer.

Table of Contents

1 How to Read This Document

A one page ‘cheat sheet’ of this document

Summary

Detailed analysis of findings

Detailed Findings

What was tested

Appendix A – Assessed Targets

The project team and schedule

Appendix B – Project Schedule

Trustwave’s test methodology

Appendix C – Test Methodology

Trustwave’s risk methodology

Appendix D – Risk Assessment

2 Priority of Weakness

This section provides the priority of the findings identified during the security assessment. The priority is based on the rated risk for each security issue.

2.1 Application Security Assessment

All the previously identified security vulnerabilities were remediated.

2.2 Infrastructure Security Assessment

All the previously identified security vulnerabilities were remediated.

2.3 AWS Infrastructure Review

Risk	Ref.	Weakness
Best Practice	AWS-1	Segmentation: Software Defined Customer Segmentation

3 Detailed Findings

This section provides detailed descriptions and analysis of the security issues identified during the security assessment of the Lexer External Facing Systems.

3.1 AWS Infrastructure Review

The following section provides an overview of the segmentation controls available between Lexer's customer's environments.

AWS-1 Segmentation: Software Defined Customer Segmentation

The Lexer AWS environment has minimal segregation of customer's environment and data using AWS resources.

- The only separation available on AWS are customer specific S3 buckets. However, the roles assigned to AWS resources have access to all S3 buckets.
- The AWS Secrets Manager contains 227 keys. However, there was no evidence that there is a unique operational / functional key for each customer.

This suggests that the segmentation is primarily software-defined. In the event the AWS environment is compromised either through a single customer user or other Internet based attack vectors, this will inadvertently result in the breach of all customer data.

Trustwave recommends the following options as best practice alternatives:

- Implement segregation of customer's environment using AWS security groups
- Where possible, implement a unique key set per customer to ensure segregation of customer PII or other sensitive data

Appendix A – Assessed Targets

As part of Lexer security assurance process, the following systems were assessed to determine the security posture of the External Facing Systems.

- Web Application Security Assessment
  - Lexer Hub - https://hub.lexer.io
- AWS Configuration Security Assessment
  - Lexer Production – https://lexer.awsapps.com/start
- External Network Security Assessment
  - account.lexer.io
  - api.lexer.io
  - assets.lexer.com.au
  - attributes-manager.camplexer.com
  - beta.lexer.com.au
  - calendar.lexer.io
  - clients.lexer.com.au
  - clients.lexer.io
  - etl.camplexer.com
  - fonts.lexer.io
  - hub.lexer.io
  - lexer.io
  - mail.lexer.io
  - nylas.lexer.io
  - sexy-asset.lexer.io
  - sexy.lexer.io
  - slack.camplexer.com
  - source-assets.lexer.io
  - status.lexer.io
  - tag.lexer.io
  - track.lexer.io
  - twitter.lexer.io
  - webhooks.lexer.io
  - www.lexer.com.au
  - www.lexer.io

Appendix B – Project Schedule

The following is the Trustwave security assessment schedule and roles and responsibilities for this engagement:

Date	Name	Role and Responsibility
26 July 2021 – 19 October 2021	Jamie Ooi	Project Management
26 July 2021 – 30 July 2021	Victor Kahan Jeremy Nunn	Technical Security Testing
4 August 2021	Jamie Ooi	Quality Assurance
19 October 2021	Troy Driver	Remediation Test
19 October 2021	Sarath Nair	Quality Assurance

Appendix C – Test Methodology

Application Testing – Test Cases

Trustwave has developed an application testing methodology that can be adapted to a range of security testing targets and with consideration of a range of industry leading benchmarks and approaches:

- Open Source Security Testing Methodology Manual (OSSTMM) v3
- SANS/MITRE Common Weakness Enumeration (CWE) Top 25
- Open Web Application Security Project (OWASP) Top 10 Vulnerabilities
- Open Web Application Security Project (OWASP) API Security Top 10
- Web Application Security Consortium (WASC)

Through building our methodology around Weaknesses rather than Attacks, we can ensure that the methodology remains relevant for a broad spectrum of system types.

We conduct our testing using a structured approach. Our testing process involves initial application familiarisation – that is, getting a thorough understanding of how the system works, how the security elements are intended to operate, and the key business logic underpinning any core transactional functionality – followed by in-depth and comprehensive assessment of the technology itself.

The test cases described below are used as a starting point for response and behaviour analysis, with the responses then used to guide subsequent phases of analysis and attack.

Our core application security testing model is based around the WASC Threat Classification view of Weaknesses. This approach allows for the key issues with web applications to be analysed, while ensuring that an 'all threats' approach is taken as to how that weakness could arise.

Ref.	Weakness	OWASP Top 10 X-Ref[1]
AW1	Application/Server Misconfiguration	2017-A6 – Security Misconfigurations 2017-A9 – Using Components with Known Vulnerabilities 2019-API7 – Security Misconfiguration
AW2	Directory Indexing	2017-A6 – Security Misconfigurations 2019-API7 – Security Misconfiguration
AW3	Improper Filesystem Permission	2017-A5 – Broken Access Control 2019-API1 – Broken Object Level Authorization 2019-API5 – Broken Function Level Authorization
AW4	Improper Input Handling	2017-A1 – Injection 2017-A4 – XML External Entities (XXE) 2017-A7 – Cross-Site Scripting (XSS)

		2017-A8 – Insecure Deserialization 2019-API8 - Injection
AW5	Improper Output Handling	2017-A1 – Injection 2017-A7 – Cross-Site Scripting (XSS) 2013-A10 – Unvalidated Redirects and Forwards 2019-API8 - Injection
AW6	Information Leakage	2017-A3 – Sensitive Data Exposure 2019-API3 – Excessive Data Exposure
AW7	Insecure Indexing	2017-A6 – Security Misconfigurations 2019-API7 – Security Misconfiguration
AW8	Insufficient Anti-automation	2017-A2 – Broken Authentication 2017-A6 – Security Misconfigurations 2019-API2 – Broken User Authentication 2019-API7 – Security Misconfiguration 2019-API4 – Lack of Resource & Rate limiting
AW9	Insufficient Authentication	2017-A2 – Broken Authentication 2019-API2 – Broken User Authentication
AW10	Insufficient Authorisation	2017-A5 – Broken Access Control 2019-API1 – Broken Object Level Authorization 2019-API5 – Broken Function Level Authorization 2019-API6 – Mass Assignment
AW11	Password Circumvention	2017-A2 – Broken Authentication 2019-API2 – Broken User Authentication
AW12	Insufficient Process Validation	-
AW13	Insufficient Session Expiration	2017-A2 – Broken Authentication 2019-API2 – Broken User Authentication
AW14	Insufficient Transport Layer Protection	2017-A6 – Security misconfigurations 2017-A9 – Using Components with Known Vulnerabilities 2019-API7 – Security Misconfiguration
AW15	Insufficient Auditing and Logging	2017-A10 – Insufficient Logging & Monitoring 2019-API10 – Insufficient Logging & Monitoring

Infrastructure Testing – Test Cases

Infrastructure security testing involves specialist consultants attempting to compromise a target system using the same techniques commonly used by malicious attackers, focused on infrastructure components such as servers, operating systems, network and security devices.

Infrastructure penetration tests are generally combined with application tests due to the significant prevalence of application level vulnerabilities and compromises originating from this source. However, infrastructure level penetration tests and vulnerability scans continue to be of value to identify misconfiguration of devices, out of date components and missing patches.

Our infrastructure security assessment process uses a 'drop in' scanning system, and runs a series of scans to identify key infrastructure security issues as detailed in the test cases below. Based on the data identified from these scans, additional testing activities may be discussed with the client to provide concrete demonstration of vulnerability and removal of false positives.

Ref.	Weakness
IW1	Software Flaws
IW2	System Misconfiguration (Servers)
IW3	System Misconfiguration (Security Devices)
IW4	Information Leakage

This usually follows the following process:

- **Network Discovery**: The purpose of this step is to discover and map out the local infrastructure of the target network. At the end of the network discovery, the penetration tester should have a basic layout of the local network infrastructure.
- **Target Identification**: This step aims to identify a host of interest. This is usually a specific IP range, or a single host/server with many available open ports and corresponding services. At the completion of the target identification step, the penetration tester would have identified a specific target that is most likely to allow penetration of the target network. This may sometimes include additional infrastructure, such additional subnets, that were discovered during the detailed assessment and analysis.
- **Vulnerability Assessment**: This step includes detailed assessment and analysis of the security posture of the identified target. This includes assessing and analysing the services and software packages running on the identified network, and vulnerabilities that are commonly found on them.
- **Vulnerability Exploitation**: The step requires that the penetration tester perform manual verifications of the vulnerabilities that are commonly found on the available services on the target system. This usually includes attempts to bypass security controls, and the lack of, to perform unauthorised and most often unauthenticated transactions with the vulnerable services identified in the previous step.
- **Network Penetration**: Successful exploitation of the identified vulnerabilities will allow unauthorised penetration of the local network infrastructure and subsequent privilege escalation activities to access sensitive data and functionality.

Security Assessment Toolset

Security assessment tools are software applications that are designed to assist in identification of security vulnerabilities, reducing the time and effort to execute repeat processes. The following tools were used during the security assessment:

- Burp Suite Pro web interception proxy
- Nessus Professional vulnerability scanner
- Nmap network security scanner
- Wireshark network analysis tool
- Nikto web application vulnerability scanner
- Sqlmap automated SQL injection auditing tool
- SSLScan SSL configuration scanner
- Recursebuster directory brute forcing tool

Time Boxing

Many applications would require an unfeasibly large amount of testing to provide coverage of all functions within the application with respect to all user types and the permutations of such users and access. This is particularly the case for systems with a high number of user types and/or privilege levels (as testing every permutation of one account's ability to interact with every other account can create hundreds, or thousands, of such permutations).

As a result, most tests are effectively "time boxed", which means that a set amount of time is allocated for testing based on the assessed risk presented by the application and the budget available, and within that time, test tasks are prioritised based on the areas of highest risk – both the most likely vulnerabilities to exist; and those that would cause the greatest harm.

Constraints

The environment provisioned for the security assessment will influence the results of the test. Where a fragile and sensitive environment is used and where network access controls are present, it may be necessary to take a 'gentler' approach to the test with a corresponding reduction in the level of coverage able to be achieved in a certain time period.

Appendix D – Risk Assessment

The ISO (International Organisation of Standardisation) 31000 series is a family of risk management standards used widely within various industries as a guideline to internal or external audit programmes. The security assessment adopts the ISO 31000 risk assessment approach, incorporating risk assessment concepts from the MITRE organisations. These form the risk ratings assessed in this report. The following tables provide description of the likelihood, consequence and resulting risk rating used in this security assessment.

The interpretation of the likelihood of an event occurring is described as per below:

Likelihood Rating	Interpretation
Almost certain	The event is expected to occur. (e.g. 1 incident every month)
Likely	The event will probably occur. (e.g. 1 incident every 6 months)
Possible	The event should occur at some time. (e.g. 1 incident every year)
Unlikely	The event could occur at some time. (e.g. 1 incident every 2 years)
Rare	The event may occur only in exceptional circumstances. (e.g. 1 incident every 5 or more years)

Trustwave considers the following as contributing factors to the likelihood of an event occurring.

- The **value** of assets contained within the vulnerable system
  E.g. Credit card details or dummy test data
- The **skills** required to successfully exploit the vulnerable system using the vulnerability identified
- The availability of **exploits** on the public domain
- The **complexity** of the exploit
- The **level of access** on the vulnerable system required to exploit the security issue
  E.g. Privileged administrative user or anonymous user

The interpretation of the consequence of an event occurring is described as per below:

Consequence Rating	Sample Interpretation
Insignificant	Little disruption to the user community. Technologies in use will require little/no effort to change. Isolated complaint from individual stakeholder able to be managed via business as usual operations.
Minor	Minor disruption to user community. The ability to provide the required service is impaired. Complaints from key stakeholder requiring management attention.
Moderate	Some inconvenience to the user community. The ability to provide a service is severely compromised. Moderate effort required to implement an alternative solution. Public criticism from key stakeholders regarding the organisation's services or activities.
Major	Noticeable impact on user community. Some core services unavailable. Potential for serious distress or minor injury. Sustained criticism from majority of key stakeholders on suitability of organisation in its current form.
Catastrophic	Community unable to function without significant support. Key technologies no longer available and no viable alternative exists. Potential for major injury or fatalities. Irreparable damage to relationships with key stakeholders and potential for organisation to cease operating in current form.

The resultant risk rating is detailed in the following risk matrix:

	Rare	Unlikely	Possible	Likely	Almost Certain
Insignificant	Very Low	Very Low	Very Low	Low	Low
Minor	Very Low	Low	Low	Low	Low
Moderate	Low	Medium	Medium	Medium	Medium
Major	Medium	Medium	High	High	High
Catastrophic	High	High	Extreme	Extreme	Extreme

Appendix E – Revision History

Version	Date	Name	Revision Comment
0.1	30 July 2021	Victor Kahan	Initial report draft
0.2	3 August 2021	Jamie Ooi	Internal report review
0.3	4 August 2021	Jamie Ooi	Client report release
1.0	19 October 2021	Sarath Nair	Final report