Example Comp. Inc.

Hybrid Penetration Test Results

Letter of Attestation

Creation Date: January 31st, 2022

Table of Contents

Introduction

This document is created as evidence and audit proof for our customer Example Comp. Inc. explaining the results of their Hybrid Penetration Test via the Intigriti platform.

Intigriti is a cloud solution, providing an ethical hacking platform to companies that desire a structured Bug Bounty & Hybrid Penetration Test.

Intigriti's Hybrid Penetration Test is built on top of the crowdsourced security platform allowing vetted security researchers to engage and communicate with companies quickly, safely, and reliably, offering live updates and communication about found vulnerabilities.

Based on the customer's predefined scope of the Hybrid Penetration Test program, a hand-picked researcher has searched for vulnerabilities and reported their findings through Intigriti's platform.

Benefits of Hybrid Pentesting

Pentest with Bug Bounty benefits
Methodology of a traditional pentest, but with the motivation, reporting, triage and rewards of Bug Bounty

Specialised skills
Hybrid penetration testers are hand-picked; selection is based on real data and criteria

Transparent researcher selection
Testers are based on previous ratings, quality, motivation, expertise, skillset

Work with experts in your field
Fintech, Retail, E-commerce, Media, Health, etc.

Highly motivated testers
Researcher receives an effort-based fee based and a capped bounty fee on top for all accepted submissions

Data-driven platform benefits
All submissions are real-time reported via the Intigriti platform

Executive Summary

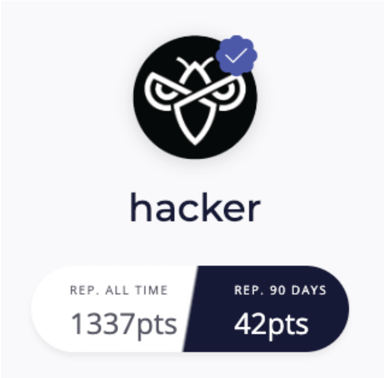
In January 2022, Example Comp. Inc. engaged Intigriti to perform a hybrid penetration test with one of Intigriti's vetted researchers eligible for pentests (see researcher information below).

The hybrid pentest was executed as a black-box assessment, meaning that no access to source code was available. The Intigriti researcher had direct 24/7 access to communication with the Example Comp. Inc. security and development team.

The tested domains and application were found to have a low security posture with 5 high to exceptional severity vulnerabilities coming up during the assessment. The key issues found would have allowed an attacker to fully compromise an admin user, gaining full access to PII data. Additionally, the integrity of all users could have been affected by changes performed via the admin account.

The Example Comp. Inc. team, together with Intigriti has identified all steps needed to remediate the found issues. Software fixes will be implemented.

Researcher information



hacker

REP. ALL TIME 1337pts

REP. 90 DAYS 42pts

Name:	hacker
Current Intigriti Ranking:	#1
Reputation All Time:	1337pts
Current Submission Streak:	Exceptional
ID Checked:	Yes

Profile: https://app.intigriti.com/profile/hacker

Scope

The scope was defined by Example Comp. Inc. and reviewed by the Intigriti team.

Assets In Scope

Domains
www.example.com/*
www.example2.com/users/*
www.subdomain1.example.com/*
www.subdomain2.example.com/*

Android Applications
com.example.androidapplication

A special test focus was requested by the Example Comp. Inc. team:

- Test if lower privileged roles can get access to admin functionality
- Test if previous cross-site scripting vulnerabilities on example.com have been remediated
- Test if authentication on Android application can be bypassed

Timeframe

This Intigriti Hybrid Penetration Test was executed during the period of January 15th 2022 to January 31st 2022. A total amount of 40 hours was performed testing all Example Comp. Inc. assets in scope.

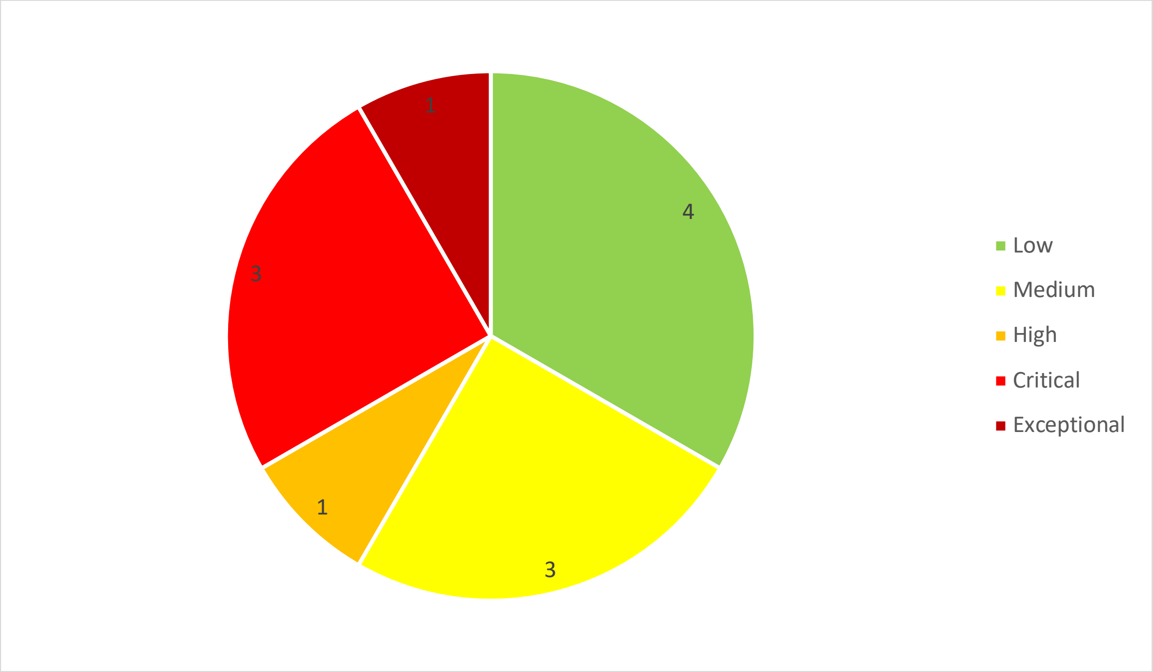
Results Summary

The high-level results of the test can be found in this document, further details can be shared at the discretion of Example Comp. Inc.

Overview

During the defined testing period, there were 12 vulnerabilities reported by our dedicated researcher: Hacker of which 12 vulnerabilities were unique and accepted in the Example Comp. Inc. Hybrid Penetration test program.

The total number of accepted vulnerabilities and their severity can be found in the chart below.



Finding Details

Type:	Vertical Privilege Escalation
Severity:	Exceptional
Date Submitted:	20.01.2022
Date Accepted:	22.01.2022
Reported Impact:	User role “viewer” can become an admin by setting “rolePermMatrix” POST parameter to 1 within user administration settings. Admin role can be used to obtain access to and change all user data.

Type:	Blind SQL-Injection
Severity:	Critical
Date Submitted:	20.01.2022
Date Accepted:	21.01.2022
Reported Impact:	An attacker can dump all data from the database.

Type:	Unauthenticated access to public MongoDB instance
Severity:	Critical
Date Submitted:	21.01.2022
Date Accepted:	22.01.2022
Reported Impact:	An attacker can anonymously login to the publicly exposed MongoDB instance to get access to all data.

Type:	Insecure Direct Object Reference
Severity:	Critical
Date Submitted:	24.01.2022
Date Accepted:	25.01.2022
Reported Impact:	An attacker can get access to any user’s personal records.

Type:	Subdomain Takeover via dangling DNS record
Severity:	High
Date Submitted:	16.01.2022
Date Accepted:	20.01.2022
Reported Impact:	Controlling the subdomain, an attacker is able to serve malicious content trusted by company users, using the domain for phishing purposes or e.g. for stealing session cookies enabling account takeovers.

Type:	Business Logic Error
Severity:	Medium
Date Submitted:	15.01.2022
Date Accepted:	17.01.2022
Reported Impact:	2FA TOTP token is not bound to user allowing an attacker to create a valid token to log in.

Type:	Improper Access Control
Severity:	Medium
Date Submitted:	25.01.2022
Date Accepted:	30.01.2022
Reported Impact:	Attacker can use all endpoints under /api/example/read/* without having the right permission group set.

Type:	Stored Cross-Site Scripting
Severity:	Medium
Date Submitted:	30.01.2022
Date Accepted:	31.01.2022
Reported Impact:	Attacker can use XSS vulnerability to post victim’s private information as a comment in thread functionality.

Type:	Broken Access Control
Severity:	Low
Date Submitted:	28.01.2022
Date Accepted:	31.01.2022
Reported Impact:	Attacker can get information if a certain email address has already registered a user.

Type:	Sensitive Data Exposure
Severity:	Low
Date Submitted:	15.01.2022
Date Accepted:	20.01.2022
Reported Impact:	Stacktrace exposes the tech stack and plugins used by the application.

Type:	Security Misconfiguration
Severity:	Low
Date Submitted:	17.01.2022
Date Accepted:	20.01.2022
Reported Impact:	CAPTCHA can be bypassed by setting setting “valid” parameter to “true”.

Type:	Path Traversal
Severity:	Low
Date Submitted:	23.01.2022
Date Accepted:	27.01.2022
Reported Impact:	Attacker can upload arbitrary files to any location.

Methodology

The security assessments by Intigriti's vetted researchers include testing for an extensive range of vulnerabilities, including those defined in the OWASP top 10 - 2021:

- A01:2021-Broken Access Control
- A02:2021-Cryptographic Failures
- A03:2021-Injection
- A04:2021-Insecure Design
- A05:2021-Security Misconfiguration
- A06:2021-Vulnerable and Outdated Components
- A07:2021-Identification and Authentication Failures
- A08:2021-Software and Data Integrity Failures
- A09:2021-Security Logging and Monitoring Failures
- A10:2021-Server-Side Request Forgery

Each submission gets triaged by Intigriti's in-house team to validate the proof of concept and ensure that the submission can be replicated.

Scoring the severity of submissions:

Intigriti's severity scoring system is based on the CVSSv3 scoring system together with business impact factors that are determined during the scoping phase of the engagement.

Appendix

Out-Of-Scope Vulnerability Classes

Web Application

Self-XSS that cannot be used to exploit other users	Verbose messages/files/directory listings without disclosing any sensitive information
CORS misconfiguration on non-sensitive endpoints	Missing cookie flags
Missing security headers	Cross-site Request Forgery with no or low impact
Presence of autocomplete attribute on web forms	Reverse tabnabbing
Bypassing rate-limits or the non-existence of rate-limits.	Best practices violations (password complexity, expiration, re-use, etc.)
Clickjacking on pages with no sensitive actions	CSV Injection
Host Header Injection	Sessions not being invalidated (logout, enabling 2FA, ...)
Hyperlink injection/takeovers	Mixed content type issues
Cross-domain referrer leakage	Anything related to email spoofing, SPF, DMARC or DKIM
Content injection	Username / email enumeration
E-mail bombing	HTTP Request smuggling without any proven impact
Homograph attacks	XMLRPC enabled
Banner grabbing / Version disclosure	Open ports without an accompanying proof-of-concept demonstrating vulnerability
Weak SSL configurations and SSL/TLS scan reports	Not stripping metadata of images
Disclosing API keys without proven impact	Same-site scripting
Subdomain takeover without taken over the subdomain	Arbitrary file upload without proof of the existence of the uploaded file

General

In case that a reported vulnerability was already known to the company from their own tests, it will be flagged as a duplicate	Theoretical security issues with no realistic exploit scenario(s) or attack surfaces, or issues that would require complex end user interactions to be exploited, may be excluded, or be lowered in severity
Spam, social engineering, and physical intrusion	DoS/DDoS attacks or brute force attacks
Vulnerabilities that are limited to non-current browsers (older than 3 versions) will not be accepted	Attacks requiring physical access to a victim’s computer/device, man in the middle or compromised user accounts
Recently disclosed zero-day vulnerabilities in commercial products where no patch or a recent patch (< 2 weeks) is available. We need time to patch our systems just like everyone else - please give us 2 weeks before reporting these types of issues	Reports that state that software is out of date/vulnerable without a proof-of-concept

Mobile

Shared links leaked through the system clipboard	Any URIs leaked because a malicious app has permission to view URIs opened
The absence of certificate pinning	Sensitive data in URLs/request bodies when protected by TLS
Lack of obfuscation	Path disclosure in the binary
Lack of jailbreak & root detection	Crashes due to malformed URL Schemes
Lack of binary protection (anti-debugging) controls, mobile SSL pinning	Snapshot/Pasteboard leakage
Runtime hacking exploits (exploits only possible in a jailbroken environment)	API key leakage used for insensitive activities/actions
Attacks requiring physical access to the victim's device	

INTIGRITI
ETHICAL HACKING PLATFORM

Contact us | Visit intigriti.com | Request a demo