KRATIKAL
SECURE FOR SURE

SECURITY TESTING

(TECHNICAL & COMMERCIAL PROPOSAL)

Document Type
CONFIDENTIAL

www.kratikal.com

To Kind Attn: Mr. Lucky Dated: 6th Nov'2024

Thank you for your interest in Kratikal's Security Testing Service; it is a pleasure providing the quotation for your process.

Please find the detailed breakdown in:

Appendix A	Scope of Work
Appendix B	Solution/Approach
Appendix C	VAPT Tools Used
Appendix D	Case Studies
Appendix E	About Kratikal
Appendix F	Major Clients
Appendix G	Timelines
Appendix H	Quotation

On receipt of your approval, you will be assigned a Project Manager who will coordinate with you throughout the process and will be responsible for ensuring the testing process is completed smoothly and to your satisfaction.

I look forward to receiving your approval.

Should you have any queries, please do not hesitate to contact:

Name	Phone No.
Paratosh Bansal (Technical)	9651506036
Shikha Garg (Commercial)	9289157332

Sincerely,
Paratosh Kumar
Kratikal Tech Pvt. Ltd.
B-70 Second Floor, Sector 67, Noida.

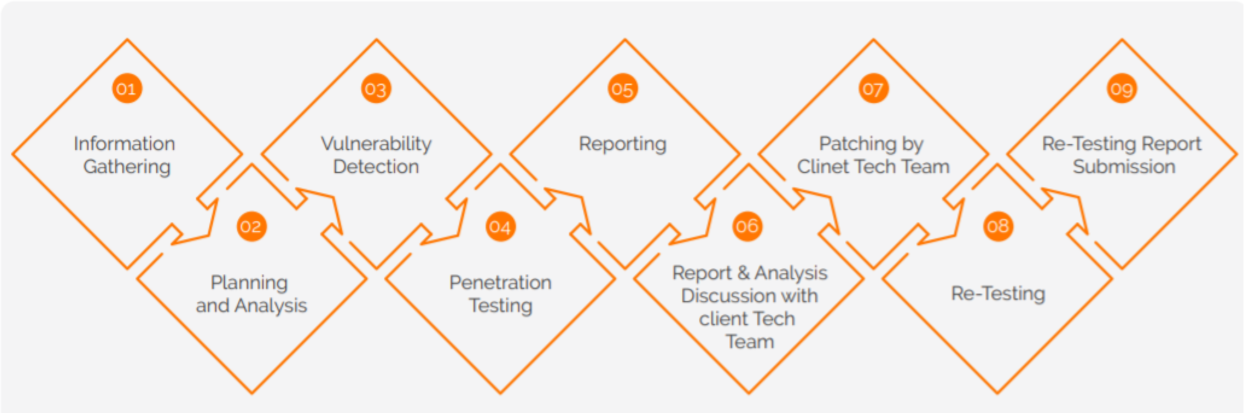
Appendix A: Scope of Work

Scope of Work – I
VAPT of Website
6 pages – 14 pages Static Website No username and password Phases: Initial Round of Testing (for 6 pages) Retesting + full Testing (for 14 pages) Additional Round of Retesting

Appendix B: Approach

Web Application Testing

We perform a comprehensive five-step approach followed by complete patching (by client) and thorough re-testing in Vulnerability Assessment and Penetration Testing methodology as illustrated in figure below:



Scope of Work

The purpose of this assessment was to evaluate the cyber security of your Web Application using simulated attacks to identify and exploit vulnerabilities in your Web Application. Malicious attacks are simulated using a variety of manual techniques supported by automated tools. Our penetration testing methodology goes beyond the detection process of simple scanning software to identify and prioritize the most vulnerable areas of your Web Application and recommend actionable solutions.

The results of this Web Application Security Testing will be used by the organization, to enhance the security feature of organization.

Approach of Work

The primary objective for a web application penetration test is to identify exploitable vulnerabilities in applications before hackers are able to discover and exploit them. Kratikal approach consists of about 70% manual testing and about 30% automated testing – actual results may vary slightly. While automated testing enables efficiency, it is effective in providing efficiency only during the initial phases of a penetration test. At Kratikal Security, it is our belief that an effective and comprehensive penetration test can only be realized through rigorous manual testing techniques.

Automated vs Manual Testing

Kratikal's approach consists of about 80% manual testing and about 20% automated testing - actual results may vary slightly. While automated testing enables efficiency, it is effective in providing efficiency only during the initial phases of a penetration test. At Kratikal Security, it is our belief that an effective and comprehensive test can only be realized through rigorous manual testing techniques.

Tools

In order to perform a comprehensive real-world assessment, Kratikal Security utilizes the hacker use oneach and every assessment. Once again, our intent is to assess systems by simulating a real-world attack and we leverage the many tools at our disposal to effectively carry out that task.

We make use of tools from the following (not a complete list):

Open source / Hacker tools (i.e.: Burp suite, Metasploit, Kali Linux, OWASP Zap, Nmap, Wireshark, Nessus, OpenVAS, Nikto, SQLMap, BeEF, Vega, Wapiti, W3a.

Web Application Penetration Testing Methodology

KRATIKAL's discursive method for web application penetration testing overlay the classes of vulnerabilities in the Open Web Application Security Project (OWASP) Top 10 2017, including but not limited to: Injection, Broken Authentication, Sensitive Data Exposure, XXE, Broken Access Control, Security Misconfigurations, XSS, Insecure Deserialization, using components with Known Vulnerabilities, and so more. Each and every web application penetration test is conducted consistently using globally accepted and industry standard frameworks. In order to ensure a sound and comprehensive application penetration test, Kratikal leverages industry standard frameworks as a foundation for carrying out penetration tests.

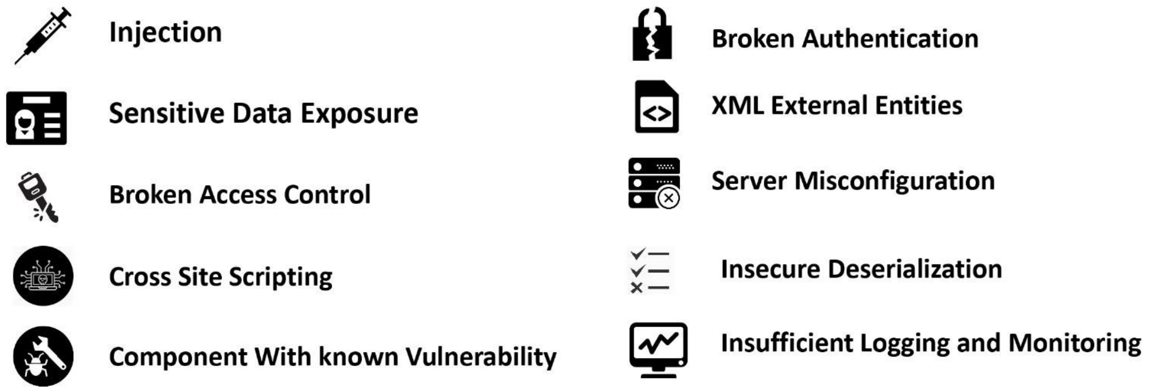


Figure 1 Open Web Application Security Project (OWASP) Top 10 Vulnerability 2017

Application Penetration Testing Steps:

Reconnaissance

The first phase in a web application penetration test is focused on collecting as much information as possible about a target application. Reconnaissance, aka Information Gathering, is one of the most critical steps of an application pen test. This is done through the use of public tools (search engines), scanners, sending simple HTTP requests, or specially crafted requests. As a result, it is possible to force the application to leak information, e.g., disclosing error messages or revealing the versions and

technologies used.

Example testing include: Conduct Search Engine Discovery and Reconnaissance for Information Leakage, Search Engine Recon, App Enumeration and App Fingerprinting, Identify app entry point

Configuration Management

Comprehending the deployed configuration of the server/infrastructure hosting the web application is nearly as critical as the application security testing itself. After all, an application chain is only as strong as its weakest link. Application platforms are wide and varied, but some key platform configuration errors can compromise the application in the same way an unsecured application can compromise the server (insecure HTTP methods, old/backup files).

Example testing includes: TLS Security, App platform configuration, File Extension Handling and Cross Site Tracing, Test HTTP strict transport security, Test HTTP methods, Test File permission.

Authentication Testing

Authentication is the process of attempting to verify the digital identity of the sender of a communication. The most common example of such a process is the log on process. Testing the authentication schema means understanding how the authentication process works and using that information to circumvent the authentication mechanism.

Example testing includes: Weak lockout mechanism, Bypassing authentication schema, Browser cache weakness, Weaker authentication in alternative channel.

Session Management

Session Management is defined as the set of all controls governing the stateful interaction between a user and the web application he/she is interacting with. In general, this covers anything from how user authentication is carried out, to what happens when they log out.

Example testing includes: Session Fixation, Cross Site Request Forgery, Cookie Management and Session Timeout, Testing for logout functionality.

Authorization Testing

Authorization Testing involves understanding how the authorization process works and using that information to circumvent the authorization mechanism. Authorization is a process that comes after a successful authentication, so the pen tester will verify this point after he/she holds valid credentials, associated with a well-defined set of roles and privileges. As a result, it should be verified if it is possible to bypass the authorization schema, find a path traversal vulnerability, or find ways to escalate the privileges

Example testing includes: Directory Traversal, Privilege Escalation and Bypassing Authorization Controls, Insecure direct object reference.

Data Input Validation

The most common web application security weakness is the failure to properly validate input coming from the client or from the environment before using it. This weakness leads to almost all of the major vulnerabilities in web applications, such as cross site scripting, SQL injection, interpreter injection, locale/Unicode attacks, file system attacks, and buffer overflows.

Example testing include: Cross Site Scripting, SQL Injection, OS Commanding and Server- Side Injection,code injection, Local file inclusion and Remote fie inclusion, Buffer overflow

Testing for Error handling

Often, during a penetration test on web applications, we come up against many error codes generated from applications or web servers. It's possible to cause these errors to be displayed by using a particular request, either specially crafted with tools or created manually. These codes are very useful to penetration testers during their activities, because they reveal a lot of information about databases, bugs, and other technological components directly linked with web applications.

Example testing include: Analysis for Error codes, Analysis for Stack Traces.

Testing for Business logic

There are many examples that can be made, but the one constant lesson is "think outside of conventional wisdom". This type of vulnerability cannot be detected by a vulnerability scanner and relies upon the skills and creativity of the penetration tester. In addition, this type of vulnerability is usually one of the hardest to detect, and usually application specific but, at the same time, usually one of the most detrimental to the application, if exploited.

Example testing include: Integrity checks, Process timing, Upload of unexpected filetype, Ability to forge request.

Clint-side testing

Client-Side testing is concerned with the execution of code on the client, typically natively within a web browser or browser plugin. The execution of code on the client-side is distinct from executing on the server and returning the subsequent content.

Example testing include: JavaScript execution, Client-side URL redirection, Cross origin resource sharing and Manipulation.

Denial-of-Service (Optional)

A denial of service (DoS) attack is an attempt to make a resource unavailable to its legitimate users. Traditionally, denial of service (DoS) attacks have been network based: a malicious user floods a target machine with enough traffic to make it incapable of servicing its intended users. There are, however, types of vulnerabilities at the application level that can allow a malicious user to make certain functionality unavailable. These problems are caused by bugs in the application and often are triggered by malicious or unexpected user input. This phase of testing will focus on application layer attacks against availability that can be launched by just one malicious user on a single machine. Not all clients have an appetite for DoS testing, therefore it may not always be a component of each and every penetration test.

Reporting

The reporting step is intended to deliver, rank and prioritize findings and generate a clear and actionable report, complete with evidence, to the project stakeholders. The presentation of findings can occur in-person–format is most conducive for communicating results. At Kratikal, we consider this phase to be the most important and we take great care to ensure we've communicated the value of our service and findings thoroughly.

Web Services / API VAPT Methodology

An API penetration test emulates an external attacker or malicious insider specifically targeting a custom set of API endpoints and attempting to undermine the security to impact the confidentiality, integrity, or availability of an organization’s resources. This methodology outlines the standards, tools used, and process that Kratikal’s Team will follow while completing an assessment according to our API penetration testing methodology.

Web Service Assessment Stages:

Our API penetration testing methodology can be broken into 4 stages:

1 Scope Information Gathering

After initiating the project target information will be collected from the client. In the case of API penetration testing, this information will include any applicable IP addresses and URLs, a definition file or documentation for all endpoint definitions, authentication credentials or API tokens (2 sets of credentials for each role being tested), and a list of any sensitive or restricted endpoints that should not be scanned or exploited.

2 Reconnaissance

The first phase will involve open-source intelligence gathering, which includes a review of publicly available information and resources. The goal of this phase is to identify any sensitive information that may help during the following phases of testing, which could include email addresses, usernames, technology in use, user manuals, open ports, etc. Additionally, this step will include searching for sensitive information that should not be publicly available, such as internal communications, salary information, or other potentially harmful information.

3

Vulnerability Analysis

The perspective of the testing will also be identified to ensure the validity of vulnerabilities discovered. This phase of the testing is also including manual approach of the vulnerable endpoints, determining business functionality of the endpoints, and identifying unauthenticated/authenticated endpoint attack surface. An application proxy will be used to capture normal API interactions for all in-scope endpoints, and packet-level traffic and response headers will be analysed. Manual identification and confirmation of vulnerabilities for each tested endpoint will be conducted, including injection-style attacks (SQL, command, XPath, LDAP, XXE, XSS), error analysis, file uploads, etc. Vulnerability identification based on identified software versions will also be attempted.

Test Cases for API Assessment

- **Broken Object Level Authorization**
- Broken User Authentication
- Excessive Data Exposure
- Lack of Resources & Rate Limiting
- Broken Function Level Authorization
- Mass Assignment
- Security Misconfiguration
- Injection
- Improper Assets Management
- Insufficient Logging & Monitoring

4 Report

The reporting step is intended to deliver, rank, and prioritize findings and generate a clear and actionable report, complete with evidence, to the project stakeholders. The presentation of findings can occur in-person–format is most conducive for communicating results.

At Kratikal, we consider this phase to be the most important and we take great care to ensure we have communicated the value of our service and findings thoroughly.

VAPT Tools Used

We use industry benchmark security testing tools across each of the IT infrastructure as per the business and technical requirements. Below are few from many of the tools we use:

- Sqlmap
- Curl
- Burpsuite
- Postman
- Swagger UI
- Nmap
- SoapUI

Appendix C : VAPT Tools Used

We use industry benchmark security testing tools across each of the IT infrastructure as per the business and technical requirements. Below are few from many of the tools we use, along with the **Manual Testing**. We can go ahead with additional add-on tools if there is any specific requirement from the client.

01

Mobile App

MobSF
Burpsuite
Xposed Framework
Dex2Jar
Drozer

02

Web App

Burpsuite
Nmap
Acunetix
Net Sparker
DIRB

03

Network & Wireless

Network Topology
Maper
Nmap
Aircrack-ng
Qualys Guard
Nessus

04

Servers

Nessus
Nmap
Metasploit
Framework
Nikto
OpenVAS

05

IoT

Shikra
Bus Pirate
JTAGulator
Facedancer21
ExplIoT

06

API Security Testing

SoapUI
Postman
Burp Suite
swagger
Fiddler

07

Desktop App Security Testing

dotPeek
Process Hacker
Sysinternals Suite
Burp Suite
CFF Explorer
Wireshark
ProcMon
dnSpy
Echo Mirage

08

Secure code review

Kiuwan

09

Docker/Container Security Testing

Docker Bench Security
Clair Scanner

10

Configuration Auditing

Nessus
CCAT Tool
Lynis

Appendix D: Case Studies

Kratikal has unequivocal quality and flawless customer experience when it comes to VAPT of IT resources. In the past 4 years, Kratikal has tested 500+ applications and 5000+ IT Infrastructures ranging from E-commerce, Fintech, BFSI, Telecom, Consumer Internet, Cloud Service Platforms, Manufacturing, Healthcare and many more. Kindly refer the link for few case studies: https://www.kratikal.com/case-studies.php

Appendix E: About Kratikal

Kratikal Tech. Pvt. Ltd. is the trusted standard for companies and individuals acquiring services to protect their brands, businesses, and dignity from baffling Cyber-attacks. We provide end to end cyber security solutions to our clients. Our thrust on securing the People-Process-Technology has enabled us to offer impenetrable security to our clients across the world. We provide complete suite of manual and automated security testing services. Kratikal provides a range of Global Expertise on cyber security services.

Team Strength: 100+

Awards and Recognitions

2021:

Kratikal is empaneled by CERT-In for providing information Security Auditing Service.

2020:

Awarded as the top cyber security startup at the 12th Top 100 CISO Awards.

2019:

Kratikal awarded amongst Top-3 Cyber Security startups at 100 CISO platform.

TAB was recognized for Excellence in Security Technology Implementation Award under the category of Detective Systems at IFSEC India Awards.

2018:

We partnered with IBM and ORACLE for assessing their cloud solutions.

Raised Pre-Series A from Yes Bank, ART Capital, Equentia and Angel Investors.

Recognized amongst Top-3 Startups in NASSCOM Product Conclave held in Bangalore.

2017:

TCOE India: We were incubated under the ministry of Telecom.

Our Flagship Product ThreatCop was recognized as 'Top-10 most innovative product" in 2017 by DSCI NASSCOM

2016:

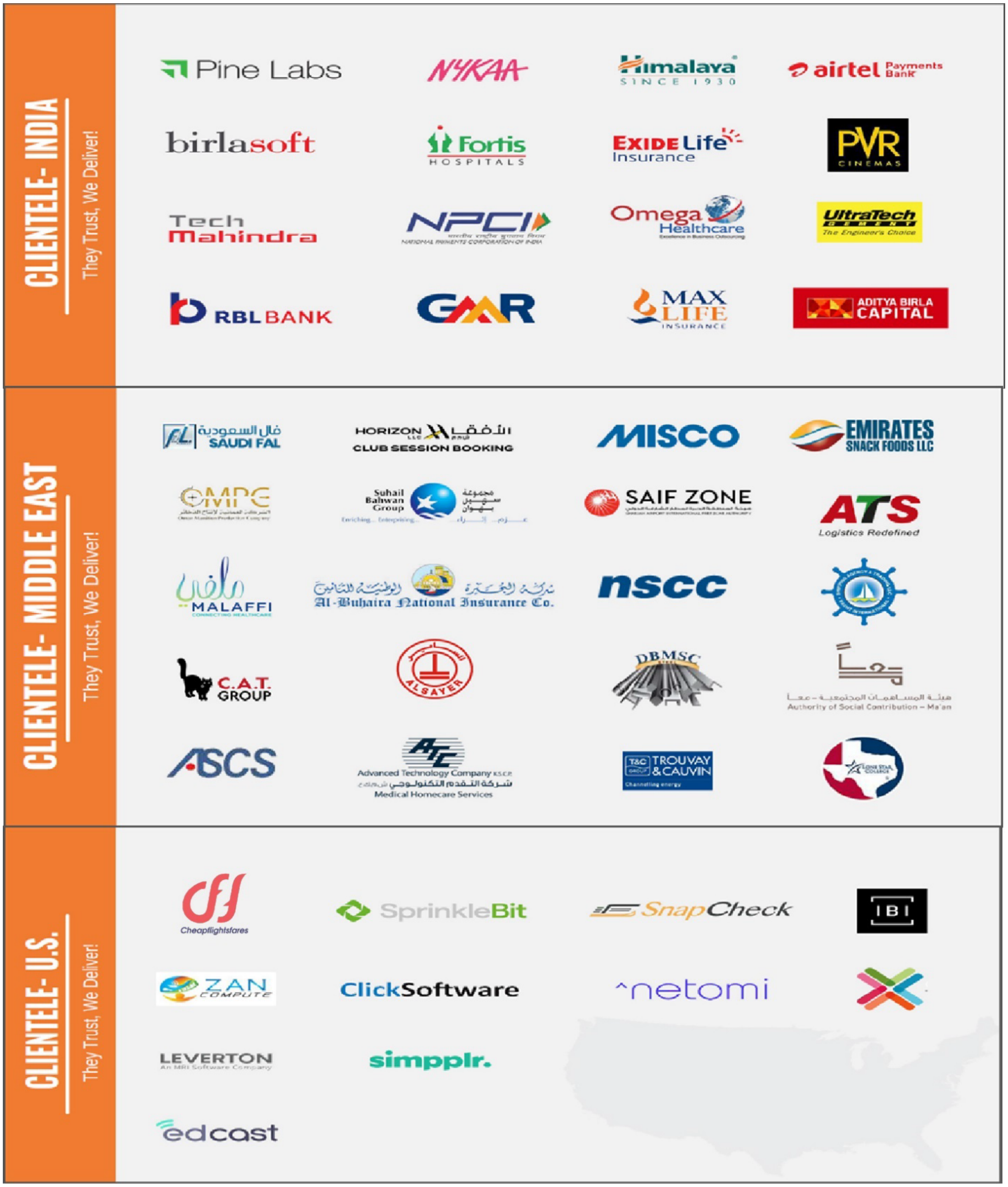
We were recognized under Startup India.

Raised Angel Investment.

2015:

We partnered with Microsoft Biz Spark

Appendix F: Major Clients



Kindly refer the link for the entire clientele: https://www.kratikal.com/clients.php

Appendix G: Timelines

Security Testing for (SOW-I)

S. No.	Milestones	Gold	Timelines
1	Milestone 1	Day 0^	Project Kick-Off
2	Milestone 2	1 Weeks^^	Security Testing as per SOW
3	Milestone 3	1 Weeks**	Security Testing + Re-testing as per SOW
4	Milestone 4	0.5 Weeks**	Additional Re-testing as per SOW

Note:

1. Re-Testing is done after the client confirms that the recommended patches have been implemented and gives go ahead with the second round of testing.
2. Security re-testing must be carried out within two months after the first testing report has been submitted.
3. In case there is no-retesting, payment shall be full.
4. ^ Day 0 is counted as the day when the kickoff cost is received from the client.
5. ^^ The timeline is as per the availability of resources.
6. **Re-testing would take place after the client gives go-ahead for the re-test.

Appendix H: Quotation

Service	Gold
Weekly coordination call with the testing team	TRUE
Access to the testing worksheet	TRUE
Executive report of testing	TRUE
Certificate of VAPT of Validity	TRUE
Automation Testing	TRUE
Manual Testing (	TRUE
Recommendations	TRUE
On call consultation post first round report submission*	TRUE
Executive presentation for managerial purposes	X
Alerts on some critical vulnerabilities and immediate remediations	TRUE
Alerts on every milestone reached	X
Free consultation for compliance like ISO, SOC2, PCI DSS	X
Recommendations on Cyber security initiatives	X
Weekly newsletter	TRUE
Multiple Re-testing on selected Vulnerabilities**	Upto 2 Rounds
Effective Pricing for One time engagement	**₹ 1,10,000**
One time special discount	**₹ 25,000**
Effective Pricing for One time engagement	**₹ 85,000**

Note:

^^ All taxes shall be charged extra at the rate decided by Government.

1. Travel, Boarding and lodging, in case to be arranged by Kratikal and if required will be charged as per actuals on top of the above price.
2. Emergency Response Team will be charged as required.
3. Commercials: 1. 50% advance payment as Project Kick-Off cost.
   2. 25% payment before the submission of first round of testing report.
   2. 25% payment before the submission of Second round of testing report.
4. PO to be raised in favor of:

Kratikal Tech Pvt. Ltd.
B-70 Second Floor
Sector 67, Noida, UP India
GSTIN 09AAFCK5333C2ZQ

Submitted to:

Submitted by:

PKumar

Paratosh Kumar (CTO),
Kratikal Tech. Pvt. Ltd.