

Enhancing Cloud Security: A Multi-Factor Authentication and Adaptive Cryptography Approach Using Machine Learning Techniques

Shwetha B¹
Assistant Professor
Department of CSE(AIML),
East Point College of
Engineering and technology,
Bangalore, India
shwetha.b@eastpoint.ac.in

Manoj R²
Department of CSE(AIML),
East Point College of
Engineering and technology,
Bangalore, India
1ep23ci027@eastpoint.ac.in

Hemanth V³
Department of CSE(AIML),
East Point College of
Engineering and technology,
Bangalore, India
1ep24ci401@eastpoint.ac.in

Karthik S⁴
Department of CSE(AIML),
East Point College of
Engineering and technology,
Bangalore, India
1ep24ci402@eastpoint.ac.in

Sai Charan S⁵
Department of
CSE(AIML), East Point College
of Engineering and technology,
Bangalore, India
1ep24ci403@eastpoint.ac.in

Vijay P⁶
Department of CSE(AIML),
East Point College of Engineering
and technology, Bangalore, India
1ep24ci404@eastpoint.ac.in

Abstract—Storing files, photos, bank records and office work on the internet (“the cloud”) is now a part of everyday life in India and across the world. This convenience also brings a danger: if a stranger gets into someone’s cloud account, that person’s private data can be stolen or misused. This paper explains, in simple words, a security design that protects cloud data in two ways. First, it checks who is logging in using several proofs together (password and PIN, fingerprint or face, a one-time code, and the way the person normally behaves). Second, it protects the data itself with locks that can change automatically when danger is sensed. A learning program (machine learning) watches every login, gives it a risk score, and decides whether to allow it, ask for more proof, or block it. The paper walks through each block of the architecture, explains how the blocks work together using a day-to-day example, compares the design with the usual “password only” method, shows the screens of a working prototype called CloudShieldAI, and discusses limits, privacy and future work. The aim is that even a reader without a technical background can look at the architecture diagram and understand how it keeps cloud data safe.

Index Terms—Cloud security, multi-factor authentication, adaptive cryptography, machine learning, anomaly detection, risk assessment, secure access.

I. INTRODUCTION

Think of the cloud as a very large, shared digital locker room that sits somewhere on the internet. Instead of keeping everything on one’s own phone or laptop, people keep it in this locker room and open it from anywhere. Students store assignments, hospitals store patient records, shops store bills, and banks store account details. India’s growth in online payments, e-governance and online education has made this habit even stronger.

The weak point is the key to the locker. For most people, the key is only a password. Passwords can be guessed, copied, reused on many websites, or stolen through fake messages that look real. Once a thief has the password, the cloud treats the thief like the real owner. In their well-known study, Sasikumar and Nagarajan [1] point out that password-only protection is no longer enough, and they propose combining several checks with a lock that can change itself.

This paper builds on that idea and explains it through the architecture picture shown in Fig. 1. The picture has four main parts, which we describe one by one:

- Multi-Factor Authentication – checking who the user is by using more than one proof.
- Machine Learning Algorithms – a “smart brain” that studies the login and spots anything unusual.
- Adaptive Cryptography – locks on the data that change according to the situation.
- Intelligent Security Response – the decision step that allows, questions or blocks the user.

We have kept the language simple and used everyday examples (a bank locker, a school gate, a house with many doors) so that a reader from outside the computer field can follow the whole story just by looking at the diagram.

A. Objectives of this paper

- To explain every block of the architecture in plain language.
- To show how the blocks work together from the moment a user types a password to the moment data is opened.
- To compare the design with the common “password only” method.
- To show the screens of the CloudShieldAI prototype that puts the design into practice.
- To discuss benefits, limits, privacy and the Indian setting.

II. BACKGROUND: CLOUD AND ITS RISKS

A. What is cloud computing?

Cloud computing means using computers, storage and software that belong to a service company and are reached through the internet. The U.S. National Institute of Standards and Technology describes it as on-demand, shared and quickly scalable service [2]. A simple comparison is electricity: we do not run our own power plant; we just plug in and pay for what we use. In the same way, a student can use cloud storage without owning any large computer.

B. Why cloud data needs extra care

Because many customers share the same cloud service and everything is reachable through the internet, an attacker does not need to break into a building; a screen and an internet connection are enough. Studies of cloud services

have long listed weak login control, data leakage and misuse of accounts among the main concerns [4]. If the login step fails, everything behind it is exposed.

C. Common ways accounts get broken into

- Guessing and brute force: a program tries thousands of passwords until one works.
- Phishing: a fake SMS, e-mail or website tricks the user into typing the password.
- Stolen or reused passwords: one leaked password opens many accounts.
- Impersonation: a person pretends to be the owner using copied details.
- Insider misuse: someone with legitimate access uses it wrongly.

D. Why a single password fails

A password is “something you know.” Anyone who learns it becomes you. Researchers who compared password systems with newer methods concluded that passwords are cheap and familiar but weak against many attacks, and that other methods bring their own trade-offs [8]. This is why modern guidelines recommend using more than one kind of proof [3].

III. THE PROPOSED ARCHITECTURE AT A GLANCE

Fig. 1 shows the complete architecture. It is best read like a story from the top to the bottom. At the top, the Authorized Users (left cloud: the actual people) want to reach the Cloud Services (right cloud: the servers, files and applications). Between them stands a big shield with a lock called the Secure Access Gateway. Nobody and nothing passes this gateway without being checked.

On the left of the gateway is the orange block, Multi-Factor Authentication, which asks “Are you really who you claim to be?” On the right is the blue block, Adaptive Cryptography, which asks “How strongly should the data be locked right now?” In the middle is the Machine Learning Algorithms block, the thinking part that connects both sides. Below them is the dark-blue bar, Intelligent Security Response, where the final decision is taken. The two bottom corners show the result: Secure Data Storage (data stays locked) and Safe User Access (the right user gets in with a green tick).

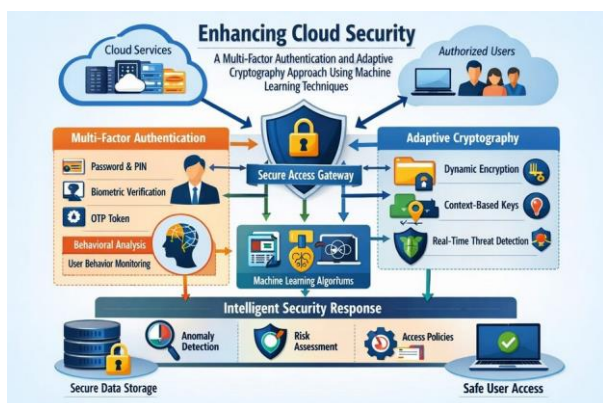


Fig 1. Overall architecture of the proposed cloud security approach: multi-factor authentication and adaptive cryptography guided by machine learning.

The sections that follow open each block of this figure and explain it with simple examples.

IV. MULTI-FACTOR AUTHENTICATION (MFA)

Multi-factor authentication means proving identity in more than one way. The idea is the same as a bank locker: the customer needs the bank’s key and also their own key; one alone cannot open it. Surveys show that using several proofs together greatly lowers the chance of a successful break-in [7]. Table 1 summarises the four checks drawn in the orange block of Fig. 1.

TABLE 1. Checks used in the MFA block.

Check	Kind of proof	Everyday example
Password & PIN	Something you know	House door key
Biometric	Something you are	Fingerprint on a phone
OTP token	Something you have	Code sent to mobile
Behaviour	How you act	Your usual typing style

A. Password and PIN

This is the first and most familiar gate. The user enters a password and a short PIN. Good practice here is to use long, unique passwords and never to store them as plain text. Servers should keep only a scrambled version of the password, so that even if the server is stolen, the real password is not visible. On its own this step is weak, which is why it is only the beginning of the process.

B. Biometric verification

Biometrics use a body feature such as a fingerprint or face to confirm identity. Researchers describe biometrics as a way of recognising people by who they are instead of what they remember [6]. Indian users are already comfortable with this, since fingerprint and face unlock are common on smartphones. A fingerprint is hard to copy and cannot be forgotten like a password. The system must still be careful against fake copies (for example, a photo shown to the camera), so it also checks that the sample is “live”.

C. OTP token

An OTP (one-time password) is a short code, usually six digits, that works only once and only for a few minutes. It is sent to the user’s registered phone or created by an app. Indian users know this well from bank and UPI transactions. Even if a thief knows the password, the thief does not hold the user’s phone, so the OTP stops the attack. The code expires quickly, so an old code is useless.

D. Behavioural analysis and user behaviour monitoring

The last box in the orange block watches how the user behaves. Everyone has habits: the usual login time, the usual city, the device they use, how fast they type and how they move the mouse or swipe the screen. The system quietly learns these habits. If someone logs in at 3 a.m. from a new device in a different country, and types in a very different way, it is a warning sign, even when the password and code are correct. This step does not trouble the real user at all, because it happens in the background.

V. THE MACHINE LEARNING BLOCK

In the centre of Fig. 1 is the block named Machine Learning Algorithms. Machine learning simply means a computer program that learns from many past examples instead of being given fixed rules. A child learns to tell a

dog from a cat by seeing many pictures; the program learns to tell a genuine login from a suspicious one in the same way.

A. What does it learn from?

- Past login records: time, place, device and number of failed tries.
- Typical habits of each user (the behaviour data from the MFA block).
- Known attack patterns, such as thousands of wrong passwords in one minute.
- Feedback: when the system is wrong, the correction is used to improve it.

B. What does it produce?

For every login the program produces a risk score, which is a number showing how suspicious the login looks (Fig. 2). A low score means “looks like the usual user.” A high score means “something is wrong.” This score is sent both to the left side (MFA, which may ask for more proof) and to the right side (cryptography, which may strengthen the locks). The original study reports that a learning model of this kind reached about 96.8% accuracy in recognising attacks in its tests [1]; this paper uses that result only to show that the idea is practical.

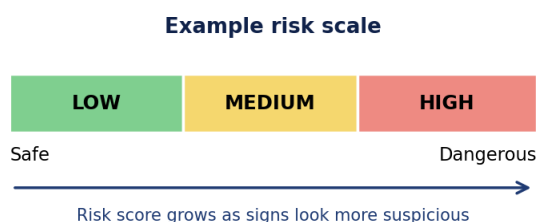


Fig 2. Example of a risk scale used by the machine learning block.

C. Why learning is better than fixed rules

Fixed rules such as “block anyone who fails three times” are easy for attackers to learn and avoid. A learning system keeps changing. It notices new tricks, adjusts to the user’s changing habits (for example, a person who moves to a new city) and reduces false alarms over time. Surveys of anomaly detection show that this approach is widely used to find unusual events in banking, networks and security systems [5].

D. Signals the engine looks at

The learning block does not read the user’s private messages or files. It looks only at a small set of signals about the login itself. Table 2 lists the main ones with simple examples.

TABLE 2. Signals used to judge a login.

Signal	What it shows	Example
Time of login	Is it a usual hour?	3 a.m. is unusual
Device	Is it a known device?	New unknown laptop
Failed tries	Is someone guessing?	Ten wrong PINs
Typing style	Does it feel like the owner?	Very slow or very fast
Data use	Is the activity normal?	Thousands of downloads

E. The learning cycle

The learning block follows a simple cycle that repeats again and again. First it learns from past records. Then it is tested on records it has not seen before, to check whether it can tell good logins from bad ones. Next it is used on live logins. Finally it is updated with the new records and with the corrections made by the administrators. Without this cycle the system would slowly become out of date as users and attackers change their behaviour.

VI. ADAPTIVE CRYPTOGRAPHY

Cryptography is the science of locking data so that only the right person can read it. “Adaptive” means that the lock is not fixed; it changes according to the situation. The blue block on the right of Fig. 1 has three parts.

A. Dynamic encryption

Encryption turns readable data into a scrambled form. In a normal system, one method is chosen once and used for years. If attackers find a way to break that method, all data becomes unsafe. In dynamic encryption, the system keeps several different locking methods ready and switches between them. The study in [1] uses five pairs of methods and picks among them depending on the danger seen. It is like a house with five different types of locks on the door: a burglar who knows how to pick one type still cannot predict which one is fitted today.

B. Context-based keys

A key is the secret needed to open the lock. In context-based keys, the key is made using the situation of the user, for example the verified fingerprint, the device and the time of access. The work in [1] derives the encryption key from the user’s fingerprint data. In simple words, the key is not stored in a drawer where it can be stolen; it is created fresh from the real user’s proof each time. A thief who copies the password still cannot create the key.

C. Real-time threat detection

The third part, drawn as a shield with a green bug, continuously watches for attacks while the user is connected. Examples are a sudden flood of requests, repeated wrong attempts, or an unusual download of a huge number of files. When a threat is sensed, the signal goes back to the machine learning block, which can immediately push the system to a stronger lock. This is similar to a security guard who tightens the checks as soon as a crowd gathers at the gate.

VII. INTELLIGENT SECURITY RESPONSE

The wide bar at the bottom of Fig. 1 is where the system decides what to do. It has three parts that work one after another.

A. Anomaly detection

An anomaly is anything that does not match the normal pattern. The magnifying-glass icon in the figure stands for this step. Examples: a user in Bengaluru who suddenly appears to log in from another continent ten minutes later; or an account that normally opens five files a day but now opens five thousand.

B. Risk assessment

Here all the signals (MFA results, behaviour, threat alarms) are combined into one risk level: low, medium or

high. Not every unusual event is an attack, so the system weighs the evidence before acting. A late-night login from the user's own city and device may be only slightly risky, while a late-night login from a new country plus a failed OTP is very risky.

C. Access policies

Access policies are the rules of the organisation, such as “doctors can see only their own patients” or “salary files need a manager's approval.” The final action follows both the risk level and these rules, as shown in Table 3 and in the flow in Fig. 3.

TABLE 3. Risk level and system action.

Risk	What the system does	Data lock
Low	Allows normal access	Standard
Medium	Asks for extra proof (another OTP or fingerprint)	Stronger
High	Blocks, alerts the owner and the administrator, locks the account	Strongest

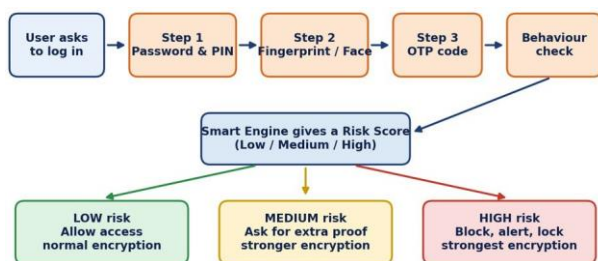


Fig 3. Step-by-step flow of a login through the proposed system.

VIII. LAYERED VIEW OF THE DESIGN

Another way to understand Fig. 1 is to see it as layers, like the many walls of a fort. An attacker must cross every layer, and each layer is built to catch what the others may miss (Fig. 4).

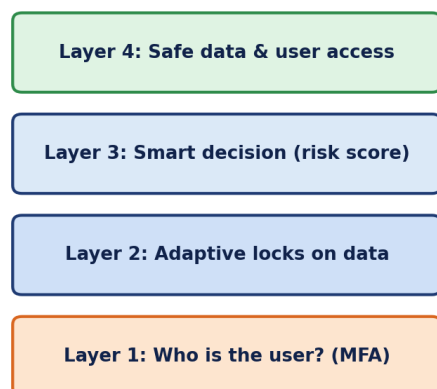


Fig 4. Layered view of the proposed security design.

The first layer checks the identity of the person. The second layer keeps the data locked in a way that can change. The third layer, the smart decision layer, watches both and gives the risk score. The fourth layer is the outcome: data stays safe and the honest user is allowed in. Because the layers are separate, a failure in one does not bring down the whole system. If a password leaks, the biometric and OTP layers still stand; if one locking method is broken, the system can switch to another.

A. How information travels

- The user sends a login request to the Secure Access Gateway.
- The gateway passes the proofs to the MFA block and the behaviour data to the learning block.
- The learning block returns a risk score to both the MFA block and the cryptography block.
- The response step compares the score with the access policy and takes the action.
- If access is allowed, the cryptography block unlocks only the data the user is permitted to see.
- Everything is recorded in a log so that mistakes can be studied and the system can improve.

IX. SECURE DATA STORAGE AND SAFE USER ACCESS

The two ends of the bottom row in Fig. 1 show the result of the whole process. Secure data storage (the stack of servers with a lock) means that the data sitting in the cloud is always kept in locked form. Even if a thief somehow reached the storage, he would see only scrambled content. Safe user access (the laptop with a green tick) means that the genuine user reaches the data easily after passing the checks, and the connection is protected from outsiders.

The aim is a balance: strong protection for the data, but no unnecessary trouble for the honest user. When the risk is low, the user may notice nothing more than a quick fingerprint touch.

X. A DAY-TO-DAY EXAMPLE

To see how everything fits together, consider Ravi, a student of a college in Bengaluru who keeps his project files on a cloud service.

- Normal day. At 9 p.m. Ravi opens his laptop at home. He types his password and PIN and touches the fingerprint sensor. The system sees the usual device, city and typing speed, so the risk is low. He gets in without further questions and his files open.
- Slightly different day. Ravi logs in from a friend's laptop in the college library. The device is new, so the risk becomes medium. The system asks for the OTP sent to his phone. After he enters it correctly, access is allowed and the files are kept under a stronger lock during that session.
- Attack attempt. A stranger who has stolen Ravi's password tries to log in at 3 a.m. from another country. The fingerprint does not match, the OTP goes to Ravi's phone (not the stranger's), and the behaviour is completely different. The risk is high. Access is blocked, Ravi gets an alert message, and the data stays locked.

In all three cases the same architecture is at work, but the system behaves differently according to the situation. That is the meaning of the word “adaptive.”

XI. COMPARISON WITH PASSWORD-ONLY SECURITY

Table 4 compares the usual method with the architecture described in this paper.

TABLE 4. Password-only versus proposed design.

Point	Password only	Proposed design
Proofs needed	One	Several, chosen by risk
If password leaks	Account open to thief	Thief still blocked
Data lock	Fixed, never changes	Changes with danger
Attack detection	Mostly after damage	Watched live
Honest user effort	Low	Low when risk is low
Learning	None	Improves with data

XII. ATTACKS HANDLED BY THE DESIGN

Table 5 lists common attacks and the part of the architecture that helps against each one. The original research [1] reports that its framework held up against attacks of these types.

TABLE 5. Threats and the block that helps.

Attack	Plain meaning	Helpful block
Brute force	Trying many passwords	Threat detection, lockout
Phishing	Fake page steals password	OTP, biometric
Spoofing	Fake identity or device	Behaviour check, biometric
Guessing	Guessing PIN or password	Multiple factors
Impersonation	Pretending to be owner	Biometric, risk score
Data theft	Stealing stored files	Encrypted storage

XIII. COMPARISON WITH OTHER SINGLE-METHOD APPROACHES

Many systems today use one strong method alone, such as only an OTP or only a fingerprint. Table 6 shows why combining methods and adding a smart decision step is more reliable.

TABLE 6. Single methods versus combined design.

Method	Good point	Weak point
Password only	Easy and cheap	Easily stolen or guessed
OTP only	Hard to reuse	Phone theft or SIM fraud
Biometric only	Hard to forget	Cannot be changed if leaked
Fixed rules	Simple to follow	Attackers learn the rules
Proposed design	Many layers, learns	Needs careful set-up

XIV. BENEFITS FOR DIFFERENT GROUPS

- Ordinary users get better protection for photos, documents and money without having to remember very complicated rules.
- Organisations lower the chance of data leaks, which saves money, time and reputation.
- Administrators receive clear alerts and risk levels instead of a long list of raw records.
- Society benefits when public services and banks stay safe and citizens trust digital India.

XV. GOOD HABITS FOR USERS

Even the best system works better when users also follow simple habits. These are easy to learn and useful for everyone.

- Use a long and different password for every important account.

- Never share an OTP with anyone, even if the caller claims to be from a bank or the company.
- Check the web address before typing a password; fake pages often have small spelling changes.
- Keep the phone and laptop updated and locked with a PIN or fingerprint.
- Log out from shared or public computers.
- Report any alert message about a login you did not make.

XVI. WHERE THE DESIGN CAN BE USED IN INDIA

The same architecture can protect many kinds of services. A few examples that are close to daily life in India are given below.

- Education. Colleges and universities keep marks, admission records and research files online. Teachers and students can log in with a password and fingerprint, while the system watches for strange logins such as a student account opening staff-only files.
- Healthcare. Hospitals and clinics store patient reports and scans. Doctors need quick access, but outsiders must never see the records. Risk-based checks allow fast access in normal conditions and demand more proof when something looks odd.
- Banking and payments. People already use OTPs for online transactions. The behaviour check adds one more silent layer by noticing a change in device, place or habit.
- Government services. Online portals for certificates, taxes and benefits hold sensitive citizen data. A strong yet easy login helps citizens in cities as well as villages.
- Small businesses. Shops and start-ups that keep accounts and customer lists in the cloud often have no security experts. A ready-made service built on this design can protect them without extra effort.

XVII. HOW THE DESIGN CAN BE BUILT

Building such a system does not have to be done in one big step. A sensible order is shown below.

- Step 1 – Strengthen the login. Start with a strong password rule, a PIN and an OTP for every user.
- Step 2 – Add biometrics. Add fingerprint or face check, using the sensors already present in phones and laptops.
- Step 3 – Collect safe records. Begin recording login time, device and place in an encrypted log, with the user’s knowledge.
- Step 4 – Train the smart engine. Use the records to teach the learning program what is normal and what is suspicious, and test it carefully before using it live.
- Step 5 – Switch on adaptive locks. Let the risk score choose the lock strength and the action. Review the decisions regularly and correct mistakes.

XVIII. RESULTS: THE CLOUDSHIELDAI PROTOTYPE

To show that the design can be put into practice, a prototype named CloudShieldAI was built. It is a small

website that protects a user's documents using the first layers of the architecture: a secure login, a one-time code, a face check and encrypted document storage. Figs. 5 to 7 show its main screens, and the paragraphs below explain each one in everyday language.

A. The home page

Fig. 5 is the front door of the prototype. It carries the tagline “Smart Security for Your Digital Documents” and a top menu with Home, Dashboard, Authenticate, About Us and Logout. Below the welcome text, three cards explain the checks a user meets. Secure Login keeps the password in a scrambled (hashed) form and never as plain text. OTP Verification sends a one-time code to the registered e-mail or phone whenever it matters. Face Verification asks for a quick face check before the account unlocks. These three cards are the “something you know”, “something you have” and “something you are” checks of Table 1.

B. The dashboard and document protection

Fig. 6 shows the dashboard, where the user protects a file. The user drags a file into the dotted box or presses Choose File. After the upload, the page confirms that the

document was protected successfully and that a secure access code has been sent to the registered e-mail or phone. The My Documents list then shows the file name, the date and time of upload, its size (304.8 KB in this example) and the word Protected, with Open and Delete buttons. In the language of the architecture, this is the Secure Data Storage block: the file sits in the cloud in locked form, and opening it later needs the fresh access code.

C. The About page

Fig. 7 is the About page, which explains the working to ordinary users. When someone logs in, the system confirms the password, sends a one-time code to the e-mail or phone, and verifies the face. Every uploaded document is encrypted, and opening it later needs a fresh, short-lived access code. The page ends with the reason for all this effort: a single stolen password should not be enough to reach the documents.

The screens above cover the identity checks and the document protection parts of the design. The risk-score engine and the adaptive choice of locks are described in Sections V to VII, and the prototype screens are shown only to demonstrate that the idea works in a real interface.

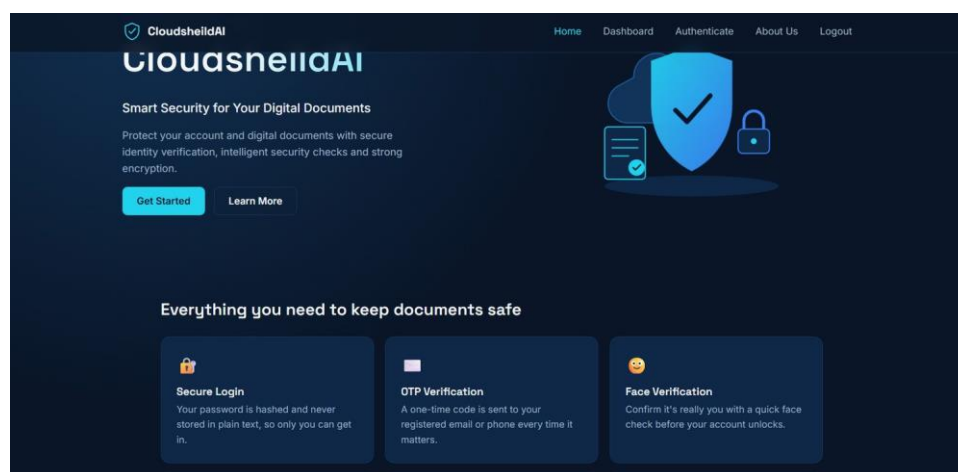


Fig 5. Home page of the CloudShieldAI prototype showing the Secure Login, OTP Verification and Face Verification features.

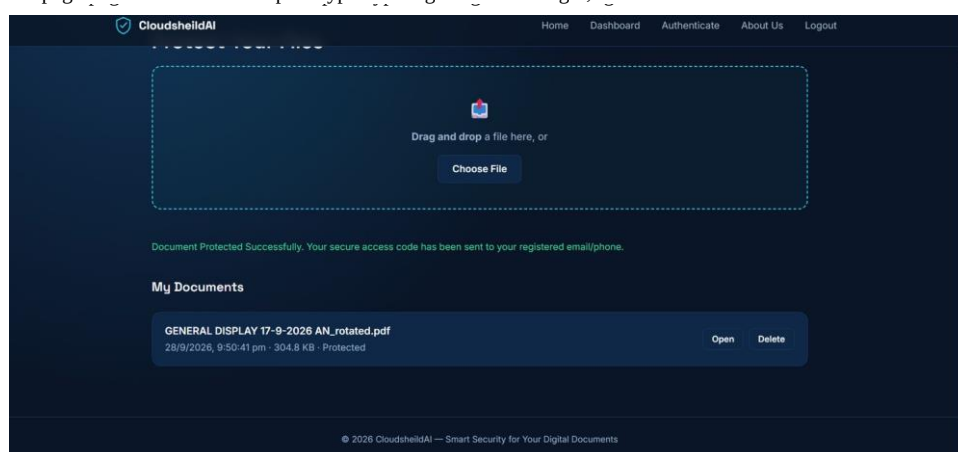


Fig 6. Dashboard of CloudShieldAI after a document is protected: the access-code message and the My Documents list.

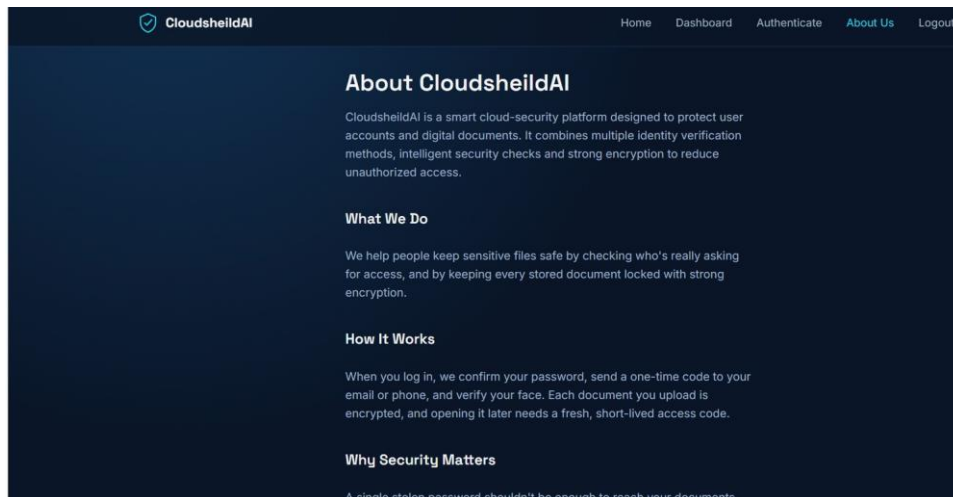


Fig 7. About page of CloudShieldAI explaining what the platform does and how it works.

XIX. HOW TO MEASURE SUCCESS

A security system should be judged by facts, not by claims. Table 7 lists simple measures that a college or company can track after the system is switched on.

TABLE 7. Simple measures of success.

Measure	What it tells us	Wish
Attack catch rate	How many real attacks were stopped	High
False alarm rate	How often honest users were stopped	Low
Login time	How long a normal login takes	Short
Response time	How fast a threat is acted on	Short
User satisfaction	How easy users find the system	High

XX. PRIVACY, LAW AND THE INDIAN SETTING

A system that studies behaviour and uses fingerprints must itself be handled with great care. Biometric and behaviour data are personal, and if they leak they cannot be replaced like a password. Good practice includes collecting only what is needed, keeping it in encrypted form, telling users what is collected, and deleting it when it is no longer required.

India has been strengthening its rules in this area. The Digital Personal Data Protection Act, 2023 sets duties for organisations that handle personal data and rights for the individual [9]. The CERT-In directions of 2022 require organisations to report certain cyber incidents quickly and to keep system logs [10]. A cloud service built on the architecture above should be planned to follow these rules from the start. Keeping data inside the country when required, and allowing the user to see and correct their data, are practical steps.

XXI. FAIRNESS AND USER TRUST

A security system that makes decisions about people must be fair and understandable. If it wrongly blocks a genuine user again and again, trust is lost. Therefore the organisation should explain in simple words what is watched and why, give an easy way to appeal when a user is wrongly stopped, and test the learning block on different groups of users (for example, people who travel often, people with older phones, and people with weak fingerprints) so that nobody is treated unfairly. Human administrators should stay in charge of serious actions such as permanently locking an account.

XXII. CHALLENGES AND LIMITATIONS

- False alarms. A genuine user who travels or changes phone may be questioned. The system should learn quickly and make the extra step easy.
- Learning needs data. A new user has no history, so the behaviour check is weak in the first days. Stronger fixed checks should be used until enough data is gathered.
- Cost and speed. Extra checks and switching of locks need computing power. The design must be tuned so that logins remain fast, even on low-cost phones and slow networks.
- Attackers also learn. Criminals try to copy behaviour or fool the learning program, so the models must be tested and updated regularly.
- Biometric safety. Fingerprint data must be stored securely, and an alternative must exist for users whose fingerprints do not read well.
- Scope of this paper. This paper explains the architecture and its working. The prototype screens in Section XVIII show that the interface works, but they are not a performance test. Numerical results quoted come from the cited study [1].

XXIII. FUTURE SCOPE

- Extending the prototype and measuring login time and user satisfaction with real college or small-business users.
- Supporting regional-language messages and alerts so that more Indian users understand security prompts.
- Using lighter learning models so that the system runs well on ordinary smartphones.
- Preparing the locking methods for future powerful computers that may break today's locks.
- Letting users see, in simple words, why they were asked for an extra check, which builds trust.

XXIV. SIMPLE GLOSSARY

Table 8 gives short meanings of the main words used in this paper, for readers from other fields.

TABLE 8. Words used in this paper.

Word	Simple meaning
Cloud	Storage and services reached through the internet
Authentication	Proving that you are who you say you are

Word	Simple meaning
Multi-factor	Using more than one proof to log in
Biometric	Fingerprint, face or other body feature used as proof
OTP	A code that works only once for a short time
Encryption	Scrambling data so outsiders cannot read it
Key	The secret needed to unscramble the data
Machine learning	A program that learns from past examples
Anomaly	Something that does not match the usual pattern
Risk score	A number showing how suspicious a login is
Gateway	The checkpoint every user must pass

XXV. CONCLUSION

The architecture explained in this paper protects cloud data from both sides. On the user side, it confirms identity using a password and PIN, a fingerprint or face, a one-time code and the user's usual behaviour. On the data side, it uses locks that change with the danger, keys created from the user's own proof, and constant watching for threats. A machine learning block connects the two sides by giving each login a risk score, and the intelligent response step turns this score into a clear action: allow, ask for more proof, or block.

The strength of the design is that no single weakness is enough to open the data: a stolen password alone does not help the thief, and a changing lock does not help an attacker who has learned one method. At the same time, the honest user is troubled only when the situation really looks risky. The CloudShieldAI prototype shows how these ideas can appear in a simple website that ordinary people can use. With careful attention to privacy, speed and Indian legal requirements, this approach can make everyday cloud use safer for students, businesses and government services alike.

ACKNOWLEDGMENT

The authors thank the Department of CSE (AIML), East Point College of Engineering and Technology, Bengaluru, for the guidance and support given during this work.

REFERENCES

- [1] K. Sasikumar and S. Nagarajan, "Enhancing cloud security: A multi-factor authentication and adaptive cryptography approach using machine learning techniques," *IEEE Open J. Comput. Soc.*, vol. 6, 2025, doi: 10.1109/OJCS.2025.3538557.
- [2] P. Mell and T. Grance, "The NIST definition of cloud computing," *Natl. Inst. Standards Technol.*, Gaithersburg, MD, USA, Special Publication 800-145, 2011.
- [3] P. A. Grassi et al., "Digital identity guidelines," *Natl. Inst. Standards Technol.*, Gaithersburg, MD, USA, Special Publication 800-63-3, 2017.
- [4] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *J. Netw. Comput. Appl.*, vol. 34, no. 1, pp. 1–11, 2011.
- [5] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surveys*, vol. 41, no. 3, pp. 1–58, 2009.
- [6] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 4–20, Jan. 2004.
- [7] A. Ometov et al., "Multi-factor authentication: A survey," *Cryptography*, vol. 2, no. 1, p. 1, 2018.
- [8] J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano, "The quest to replace passwords: A framework for comparative evaluation of web authentication schemes," in *Proc. IEEE Symp. Security Privacy*, 2012, pp. 553–567.

- [9] Government of India, "The Digital Personal Data Protection Act, 2023," Ministry of Law and Justice, New Delhi, India, 2023.
- [10] Indian Computer Emergency Response Team (CERT-In), Ministry of Electronics and Information Technology, "Directions under sub-section (6) of section 70B of the Information Technology Act, 2000 relating to information security practices, procedure, prevention, response and reporting of cyber incidents," New Delhi, India, Apr. 2022.