Interdisciplinary Journal of Global and Multidisciplinary
IJGAM
ISSN xxxx-xxxx
+62852-3430-9925 https://jurnal-ijgam.or.id/ admin@jurnal-ijgam.or.id
Diterbitkan Oleh PT. RADJA POS MEDIA
Vol. 2, No. 1, January 2026 | Page. 753 - 762

DESCRIPTIVE EXPLORATION OF THE USE OF CRYPTOGRAPHY TO PROTECT DATA ACROSS DIFFERENT FILE SIZES

Akmal Baihaqi[1], Ibnu Ruysdi[2], Intan Widya Saputri Nst[3]
Faculty of Science and Technology,
State Islamic University of North Sumatra, Indonesia
Email: akmalbaihaqi4486@gmail.com, ibnurusydi@dharmawangsa.ac.id, intanwidyasaputrinst@gmail.com

Keywords:
Cryptography,
Encryption,
File Size,
Data Protection

Abstrak

The increasing volume and diversity of digital files have heightened the need for effective data protection mechanisms. Cryptography, particularly encryption, is widely recognized as a fundamental control for safeguarding data confidentiality. However, in practical environments, the way encryption is applied often varies depending on file size, usability considerations, and workflow requirements. This study aims to descriptively explore the use of cryptography to protect data across different file sizes small, medium, and large using a qualitative descriptive method based on an internet literature study. Data were collected from authoritative standards, prior academic research, and reputable industry reports available online. The analysis focused on identifying common patterns of encryption usage, implementation approaches, and practical considerations related to file size. The findings indicate that small files are typically protected through simple, user-driven encryption methods to support ease of sharing, while medium-sized files are often bundled into encrypted containers for efficiency. Large files, such as backups and system images, tend to rely on system-level or automated encryption solutions to ensure scalability and consistency. Overall, the study highlights that implementation choices, usability, and key management practices are more influential than algorithm selection alone. This descriptive exploration provides practical insights for students, practitioners, and organizations seeking to apply cryptography effectively in real-world file protection scenarios.

INTRODUCTION

In modern information systems, files ranging from small text documents to large multimedia archives are routinely stored, transmitted, and shared across heterogeneous platforms. This everyday reality increases exposure to unauthorized access, leakage, or manipulation, especially when files contain personally identifiable information, business records, educational data, or confidential communications. Cryptography, particularly encryption, remains one of the most widely recognized technical controls for protecting confidentiality and supporting integrity in digital environments. Guidance from the U.S. National Institute of Standards and Technology (NIST) frames encryption as a core mechanism for protecting data both "during transmission and while in storage," emphasizing that cryptographic choices should align with security objectives and operational constraints.

A key practical issue that emerges in real-world adoption is the relationship between file size and encryption use. While the security goal preventing unauthorized reading does not change with file size, user experience and operational cost can. For example, encrypting a 50 KB document may be nearly instantaneous on modern hardware, while encrypting a multi-gigabyte backup can influence processing time, storage workflows, and system performance. These tradeoffs affect how individuals and organizations decide *when* to encrypt, *what* tools to use, and *how* to integrate encryption into standard processes such as archiving, sharing, cloud synchronization, and backup.

Prior research has frequently examined encryption performance using quantitative benchmarks (time, throughput, CPU usage). For instance, comparative studies have evaluated symmetric algorithms (including AES and Blowfish) across block sizes and data types, focusing on encryption/decryption speed and efficiency. Other experiments isolate how file size affects performance under different AES modes (e.g., ECB, CBC, GCM) across small to larger sample sizes, showing that measurement outcomes can vary by mode and test setup. These studies are valuable for engineering optimization, yet they often leave open a simpler, user-facing question: How do people actually use cryptography to protect files of different sizes, and what practical patterns appear in everyday contexts?

This study adopts a qualitative descriptive approach using an internet-based literature review to explore how cryptography is used to protect data across different file sizes. Rather than aiming to prove one algorithm is universally faster or "better," the emphasis is on mapping and interpreting common practices: (1) small files (documents, spreadsheets, PDFs), (2) medium files (photo collections, course materials, short videos), and (3) large files (backups, virtual machine images, archives). The approach is motivated by the observation that file-size differences often lead to different user behaviors: some users rely on full-disk encryption for everything; others selectively encrypt archives; still others use encrypted cloud links for sharing.

The relevance of this topic is reinforced by recent reporting on breach impacts and the evolving threat environment. IBM's Cost of a Data Breach Report 2025 reports that the global average breach cost declined to USD 4.44 million (the first decline in five years), while also highlighting the role of faster containment driven by AI-powered defenses and the reality that attackers are also leveraging AI. Even if breach cost fluctuates year to year, the persistent financial and operational consequences underscore why encryption remains a central control especially for data-at-rest and data-in-transit. In practice, encryption can reduce the utility of stolen files, though overall risk

reduction depends on implementation quality (key management, access control, and governance).

Cryptographic standards provide a stable foundation for these implementations. The Advanced Encryption Standard (AES) is a long-established symmetric encryption standard used globally in many products and protocols. However, standards alone do not determine outcomes; the "last mile" decisions tool selection, file packaging, password practices, and whether encryption is applied per-file, per-folder, per-archive, or at full-disk level shape real protection.

Therefore, this study aims to produce a descriptive understanding that can be useful to students, practitioners, and organizations seeking pragmatic guidance. Specifically, it explores:

1. What encryption approaches are commonly used across file sizes
2. How usability and workflow influence encryption choices
3. What tradeoffs are reported in the literature regarding large-file handling; and
4. How guidance from standards bodies connects to everyday practices.

The expected contribution is a structured synthesis that bridges "formal cryptography" and "real-world file workflows," offering insights that are simpler than performance-centric comparisons yet still grounded in credible sources and current security context

LITERATURE REVIEW

Cryptography for Data Protection: Core Concepts and Objectives

Cryptography is a foundational discipline for protecting digital information, primarily through confidentiality, integrity, authentication, and non-repudiation. In the context of file protection, the most direct objective is confidentiality, typically achieved through encryption that transforms readable plaintext into ciphertext that is computationally infeasible to interpret without the key. NIST guidance emphasizes the use of cryptographic mechanisms to protect sensitive information while stored and transmitted, positioning encryption as a practical safeguard when implemented correctly.

Encryption approaches relevant to files usually fall into symmetric and asymmetric categories. Symmetric encryption uses the same key for encryption and decryption and is widely used for bulk data because it is efficient for large content. Asymmetric encryption uses a public/private key pair and is often used to protect symmetric keys or enable secure sharing without a shared secret. In many everyday file-protection tools, a hybrid model appears: a random symmetric key encrypts the file content, and that key is protected via a password-derived key or a recipient's public key.

The AES standard is central in modern symmetric cryptography. As a U.S. federal standard, AES specifies a symmetric block cipher used broadly across security applications and is frequently implemented in file encryption, secure storage, and communication protocols. While AES is a standard algorithm, security outcomes depend heavily on mode of operation (e.g., CBC, GCM) and on implementation details like key derivation, randomness, and authentication mechanisms. Importantly, file size does not change the theoretical strength of AES, but it can affect practical choices: users may prefer full-disk encryption for simplicity, or per-file encryption for selective sharing.

From a practical protection standpoint, encryption is most effective when combined with sound key management and access controls. Without secure handling of

keys and credentials, encryption may provide a false sense of security. Thus, a descriptive study of cryptography use across file sizes must consider not only "what algorithm," but also "how encryption is applied" and "how keys/passwords are managed" in real workflows.

Standards and Guidance for Using Cryptography in Practice

Standards bodies provide a baseline for acceptable cryptographic practice, especially in regulated or high-assurance contexts. NIST publications, for example, offer structured guidance on how cryptographic standards should be used to protect digitized information during transmission and storage. This type of guidance matters for file protection because it encourages consistent choices strong algorithms, appropriate cryptographic services (confidentiality and integrity), and correct operational use.

AES, standardized in FIPS 197, is a well-established reference point for confidentiality protection, and it is commonly adopted as a "default" symmetric encryption algorithm across platforms. However, file encryption decisions extend beyond selecting AES: users choose tooling (operating-system features, archivers, dedicated encryption apps), decide whether encryption should be transparent (disk-level) or explicit (file-level), and determine how to share or store keys. Guidance documents stress that cryptographic mechanisms must match the security requirements and threat model, which in file workflows translates to questions like: Is the risk mainly device loss? Is it unauthorized cloud access? Is it insider misuse?

In many organizational environments, standards-based thinking also influences policy: encryption may be mandated for sensitive datasets, for backups, or for portable media. Large files (e.g., backup images) often fall under storage and disaster recovery policies, where encryption must coexist with operational constraints like backup windows and restore time objectives. A qualitative perspective is helpful here because standards provide "what," but organizations must decide "how" under real constraints budget, user capability, infrastructure, and workflow.

Finally, guidance implies that cryptography should not be treated as a single toggle. Strong encryption can be undermined by weak password practices or mishandled keys. Therefore, practical use depends on governance: training, procedures, and controls for key custody. This is particularly relevant when files grow in size and are more likely to be processed via automation (batch encryption, scheduled backups), increasing the importance of systematic key management and access control alongside encryption itself.

File Size and Encryption: What Prior Research Emphasizes

Prior research examining file size and encryption often focuses on performance implications. Comparative analyses across encryption algorithms and configurations test encryption/decryption speed under varying data sizes. For example, studies comparing AES, Blowfish, DES/3DES, and others frequently run experiments on different block sizes or file sizes to evaluate efficiency tradeoffs. These works commonly conclude that implementation, mode selection, and environment (hardware/software) can significantly affect results, meaning "fastest" can depend on context.

Some research specifically targets how file size influences AES performance across modes. An example study evaluates multiple AES modes (e.g., ECB, CBC, GCM) on different data sizes and formats, illustrating that measured encryption time can scale with file size and vary by mode. While these quantitative findings are important, they do not automatically translate into everyday choices; many users do not

pick an AES mode manually, but rather adopt tools that embed secure defaults.

Other studies evaluate encryption across file types (documents, images, audio/video) and note that data type and file handling can influence performance or workflow. A practical implication is that file size is not the only factor file type matters because it affects compression behavior, sharing patterns, and storage location. Large multimedia files are often stored in cloud drives; small documents may be emailed; archives may be zipped; backups may be automated.

Therefore, although prior work offers important benchmark insights, a qualitative descriptive study can complement it by examining the practical reasoning behind adoption: convenience, tool availability, sharing needs, perceived risk, and policy requirements. In practice, many people protect "all files" via full-disk encryption, but selectively protect "shareable files" via encrypted archives or password-protected containers choices shaped less by raw speed metrics and more by workflow fit.

Current Context: Why Encryption Use Still Matters

The motivation for encryption is strengthened by ongoing breach risk and evolving attacker capabilities. IBM's Cost of a Data Breach reporting indicates that breach costs remain substantial globally, and in 2025 the global average cost was reported as USD 4.44 million, with trends shaped by detection/containment speed and the expanding role of AI for both defenders and attackers. This reinforces that even when organizations improve response, the underlying threat environment persists.

From a file-protection perspective, encryption is a core control because it can reduce the value of data if it is stolen especially for files at rest (laptops, removable media, backups) and files stored in shared environments. Yet the practical application of encryption often collides with usability: users may skip encryption on small files because it feels "too much effort," or avoid encrypting large files because it slows transfers and backups. This usability-security tension is a central theme for a descriptive exploration across file sizes.

Additionally, modern environments are increasingly hybrid: files move between endpoints, cloud storage, collaboration tools, and backup systems. This increases the number of places where data may be exposed, and it encourages layered protection strategies full-disk encryption to mitigate device theft, encrypted archives for sharing, and encrypted backups for disaster recovery. Standards-based guidance supports this layered approach by highlighting the need to protect data during storage and transmission using appropriate cryptographic mechanisms.

In short, the current context suggests that encryption remains necessary, but adoption is shaped by practical constraints. That makes an internet-based qualitative synthesis valuable: it can identify how file size influences the way cryptography is used, how users manage tradeoffs, and which approaches appear most common and feasible in real workflows.

RESEARCH METHOD

This research uses a qualitative descriptive method with an internet-based study design. The goal is to describe and organize patterns of cryptography usage for protecting files of different sizes, rather than to measure performance experimentally. Data sources include:

1. Official standards and guidance documents (e.g., nist publications)
2. Peer-reviewed or publisher-hosted research articles and book chapters discussing

encryption practices or algorithm comparisons, and
3. Reputable industry reporting that contextualizes encryption as a control within breach prevention and impact reduction.

The data collection process followed three steps. First, relevant sources were identified through targeted keyword searches focusing on (a) encryption for data at rest and in storage, (b) file size considerations, and (c) common symmetric encryption algorithms and modes. Second, sources were screened for relevance and credibility. Priority was given to authoritative standards guidance (NIST), publisher platforms (e.g., Springer), and widely cited industry reports (IBM). Third, included sources were coded using a simple thematic approach: statements were grouped into themes such as "encryption scope (disk vs file)," "workflow integration," "file packaging and compression," "key/password management," and "large-file constraints."

Analysis was conducted via descriptive synthesis. Instead of statistical aggregation, the study compares and contrasts reported practices, recommendations, and recurring rationales. The findings are presented by file-size category (small/medium/large) and by major practical decisions (tooling approach, encryption location, usability considerations). This approach fits the study purpose because encryption adoption is shaped by human and organizational factors habits, convenience, policies, and perceived risk not only by algorithmic speed.

To improve trustworthiness, the study triangulates across multiple source types: standards guidance clarifies "what should be done," research literature illustrates "what is tested and discussed," and industry reporting highlights "why it matters now."

RESULT AND DISCUSSION

Small Files: Documents and Everyday Work Products

For small files (e.g., text, PDFs, spreadsheets), encryption usage tends to prioritize convenience and shareability. A common pattern in the literature is tool-mediated encryption: users rely on built-in operating system features (device/disk encryption) or simple file-level protections (password-protected documents, encrypted archives). Small files are frequently shared via email or messaging, which makes "per-file" encryption attractive when confidentiality is needed for a specific transfer.

From a standards perspective, the underlying cryptography often involves AES-based encryption, though end users may not see those details. AES is a widely standardized algorithm for protecting electronic data, and it is embedded across many software products and secure file-handling tools. The practical decision for small files is typically not "AES vs Blowfish," but rather "What is the simplest secure method that recipients can open?"

Small files also reveal a frequent weakness: password practices. People often reuse passwords or select weak ones for convenience, which undermines encryption benefits. This is where qualitative findings emphasize behavior: users may be willing to encrypt small sensitive files, but only if it does not introduce friction. Therefore, tools that integrate encryption seamlessly (e.g., transparent encryption at rest) often produce better compliance than tools requiring manual steps.

Another observed pattern is selective encryption: users encrypt only files that are clearly sensitive (grades, IDs, finance records), while leaving routine documents unencrypted. This behavior suggests that for small files, perceived sensitivity—not file size alone drives encryption. File size mainly affects the perceived effort: small files are

easy to encrypt and share, so encryption is feasible, but may be inconsistently applied without clear policy or habit.

Medium Files: Collections, Course Materials, and Mixed Media

Medium-sized files (tens to hundreds of MB) often include photo sets, slide decks with embedded media, short videos, and compressed folders used for submission or distribution. Here, encryption practices frequently shift toward containerization: users bundle multiple items into a single archive or container, then encrypt once. This reduces management complexity (one encrypted object instead of many) and can simplify sharing.

Research discussing encryption performance across data types indicates that encryption workflows can vary when handling images, audio, and video, even if the core algorithm is similar. In practical terms, medium files are large enough that transfer time and storage matter, but small enough that end-user tools remain viable without specialized infrastructure.

A recurring theme is compression-before-encryption. Users often compress folders first (to reduce size and combine files), then encrypt the archive. The rationale is practical: encrypted data is typically not compressible, so compressing after encryption is inefficient. This workflow is common in guides and tool usage patterns found online.

In organizational settings, medium files may live in cloud collaboration spaces. This can lead to a layered approach: (1) rely on platform access control, (2) encrypt particularly sensitive bundles before uploading, and (3) protect keys/passwords separately. Standards guidance supporting encryption for stored data aligns with this approach, but the qualitative finding is that users adopt it mainly when sharing extends beyond trusted boundaries (external partners, public links). Thus, for medium files, file size encourages bundling and structured sharing. Encryption decisions are strongly influenced by the need to balance confidentiality with recipient usability and transfer efficiency.

Large Files: Backups, Archives, and System Images

Large files (multi-GB) include system backups, disk images, large datasets, and long-form video archives. In this category, encryption usage often becomes infrastructure-driven rather than user-driven. Instead of encrypting each file manually, organizations commonly implement encryption through storage systems, backup software, or full-disk encryption. The practical reason is scale: manual file-level encryption does not fit large, automated backup workflows.

Performance research helps explain why: encryption/decryption time scales with data volume, and different configurations can influence throughput. Studies analyzing AES modes across increasing data sizes illustrate that processing time and operational impact can vary, especially when files become larger and workflows rely on automation. However, qualitative synthesis suggests that many organizations accept the overhead as a necessary cost for confidentiality, especially for portable drives and offsite backups.

Large-file encryption also amplifies the importance of key management. If encryption keys are lost, large backups become unusable, turning a security measure into an availability risk. Therefore, operational controls key escrow (where appropriate), secure storage of recovery keys, and role-based access are crucial companions to encryption. This aligns with standards-based guidance that cryptographic mechanisms must be selected and used appropriately in practice.

Another pattern is encryption at rest by default: many systems encrypt storage volumes automatically, so large files inherit protection without per-file decisions. This reduces user friction and improves coverage. The tradeoff is that sharing large encrypted files outside the system may still require additional steps (e.g., exporting to an encrypted container). Overall, for large files, file size pushes encryption toward system-level solutions, emphasizing automation, policy, and recoverability more than manual user actions.

Algorithm Choice vs. Implementation Choice: What Matters in Practice

Although many studies compare algorithms (AES vs Blowfish vs others), real-world file protection often depends more on implementation choices: encryption mode, authentication (ensuring integrity), key derivation from passwords, and secure defaults. Comparative research reviewing multiple algorithms across varying block sizes highlights that performance and efficiency can differ by algorithm and configuration. Yet for a descriptive study, the key insight is that most users do not directly choose algorithms; they choose products and workflows.

AES is a common default due to standardization and widespread support. This ubiquity increases interoperability (recipients can decrypt using common tools) and tends to steer practice toward AES-based solutions without explicit user decision-making. Conversely, older algorithms such as DES are primarily relevant historically or for legacy contexts, and modern guidance generally encourages using current, strong cryptographic standards.

For different file sizes, implementation determines usability:

- For small files, “right-click encrypt” or password-protected archives reduce friction.
- For medium files, encrypted containers simplify bundling.
- For large files, disk/volume encryption and encrypted backups reduce manual work.

This suggests that the most practical “decision variable” is not the algorithm name, but the encryption scope (disk, folder, archive, file) and key handling method (password-based vs key-based, centralized vs personal). Standards guidance supports choosing mechanisms appropriate to the system’s needs and emphasizes correct cryptographic use. Therefore, an applied conclusion emerges: for file protection across sizes, usability and secure defaults are often stronger predictors of consistent encryption use than theoretical performance differences between popular symmetric algorithms.

Linking Encryption to Risk: The Breach-Cost Motivation

Finally, the internet literature consistently frames encryption as a risk-reduction control in a broader security strategy. Industry reporting shows that breach events remain costly, and IBM’s 2025 reporting highlights both improved containment and the growing role of AI in the attacker-defender landscape. In such a context, encryption helps reduce the impact of data exposure particularly if stolen files remain unreadable.

However, qualitative synthesis emphasizes that encryption is not a standalone guarantee. If attackers obtain keys (through phishing, malware, poor key storage), encryption may fail. This becomes more pronounced when file size drives centralization: large-file encryption often relies on centralized systems and stored keys, which become high-value targets. Therefore, encryption’s protective value increases when paired with strong authentication, least privilege, monitoring, and incident response.

File size also affects risk priorities. Small files often contain high-value personal data (IDs, grades, financial records) and are easily exfiltrated, making selective encryption valuable. Large files (backups) can contain entire systems; losing an unencrypted backup can be catastrophic. Thus, many policies emphasize encrypting backups and portable drives, aligning with standards guidance for protecting stored data. In practical terms, breach-cost narratives motivate policy, but day-to-day compliance depends on workflow. The descriptive finding across sources is consistent: encryption adoption is highest when it is (1) automated, (2) minimally disruptive, and (3) supported by clear procedures for recovery and sharing. This aligns the "why" of breach impact with the "how" of file-size-sensitive implementation choices.

CONCLUSION

This qualitative descriptive, internet-based study explored how cryptography especially encryption is used to protect data across different file sizes. The synthesis suggests that file size influences how encryption is applied more than whether encryption is valuable. For small files, encryption is often selective and sharing-oriented, relying on easy tools such as password-protected archives or built-in protections. Medium files frequently drive bundling into encrypted containers to reduce management complexity and improve shareability. Large files shift encryption toward system-level solutions (full-disk encryption, encrypted backups) where automation and policy are necessary for scalability and consistency. Across sizes, the most important practical determinant is implementation: encryption scope, secure defaults, and key management. Standards-based guidance emphasizes protecting data during storage and transmission with appropriate cryptographic mechanisms, while AES remains a widely adopted foundation due to standardization and broad support. Finally, current breach-cost reporting reinforces the continued importance of encryption within broader security governance, especially as threats evolve with AI.

BIBLIOGRAPHY

Aprilia, N. F., Mafa, D., Muchtar, A. R., Rohim, K. A. A., & Firdaus, R. U. N. I. (2023). Penerapan Algoritma AES untuk Enkripsi pada Halaman Register serta Penerapan AES untuk Deskripsi pada Halaman Login Website. *Journal of Informatics Development*, *1*(2), 75-82.

Firdaus, D., & Dafy, M. Z. (2024). Peningkatan Keamanan dan Privasi Aplikasi Website DNA Sequencing Menggunakan Enkripsi AES 256 dan Query Parameterization. *Simpatik: Jurnal Sistem Informasi dan Informatika*, *4*(2), 79-88

Jati, M. A. H., Chrisnanto, Y. H., & Abdillah, G. (2024). PENGAMANAN DATA EKSTENSI FILE PDF DENGAN ALGORITMA AES DAN OTP. *Jurnal Informatika Teknologi dan Sains (Jinteks)*, *6*(3), 648-658.

Kautsar, A., & Ikhsan, M. (2025). Implementation of the Advanced Encryption Standard (AES) Algorithm and Bit Plane Complexity Segmentation (BPCS) Steganography Technique for Enhancing Text File Security. *SISTEMASI*, *14*(2), 956-968.

Maulana, H., Tahir, M., Farrohah, N., Maulana, F., & Aprilianyani, R. R. (2025). PERANCANGAN SISTEM KEAMANAN FILE MENGGUNAKAN HYBRID ENCRYPTION UNTUK PERLINDUNGAN DATA. *Jurnal RESTIKOM: Riset Teknik Informatika dan Komputer*, *7*(1), 87-96.

Nurrochim, A. (2025). Implementasi Kriptografi AES dan Steganografi Untuk Keamanan Data Customer dan Transaksi di PT Guna Bangun Jaya (LEMKRA). *Journal of Syntax Literate*, *10*(10).

Pakan, P. V., & Soetanto, H. (2025). IMPLEMENTASI KRIPTOGRAFI UNTUK MELINDUNGI INFORMASI TRANSAKSI PADA E-COMMERCE MENGGUNAKAN METODE CAESAR CIPHER DAN RC4. *JIFOSI*, *6*(2), 37-47.

Raharjo, T. (2025). *Analisis Penerapan Metode Enkripsi AES dan Kompresi LZMA untuk Keamanan Dokumen Medis Elektronik* (Doctoral dissertation, Universitas Islam Indonesia).

Ramadhani, D. E., Sabrina, P. N., & Ashaury, H. (2025). Implementasi Algoritma Enkripsi Blowfish dan Rijndael dalam Pengamanan File Text. *Jurnal Locus Penelitian dan Pengabdian*, *4*(8), 7534-7544.

Siagian, A. A., & Indra, Z. (2025). ANALISIS TEKNIK PLAYFAIR DAN SHIFT CIPHER SEBAGAI METODE KRIPTOGRAFI KLASIK UNTUK KEAMANAN DATA. *Jurnal Komputer dan Teknologi*, *4*(1), 13-19.

Simanjuntak, V., Silalahi, D. K., & Pasaribu, Y. P. (2025). Hybrid Cryptosystem dengan Algoritma 3DES dan AES dalam Pengamanan File Text. *JURNAL QUANCOM: QUANTUM COMPUTER JURNAL*, *3*(2), 26-34.

WIDYANTO, A. (2024). IMPLEMENTASI KRIPTOGRAFI TEKS MENGGUNAKAN RSA. *Community Service Articles*, *1*(2), 70-81.