Modeling Human-Behavior Effects on Ransomware Resilience with Stage-Wise Markov Chains

Fred Kembamba[1][0009−0003−2880−8757] and Javed I. Khan[1][0000−0002−7457−3368]

Department of Computer Science, Kent State University, Kent, Ohio
{fkembamb@kent.edu,javed@kent.edu}

Abstract. Human-targeted ransomware campaigns exploit phishing, credential misuse, inconsistent patching, and uneven administrative practice. This paper presents a human-behavior-aware ransomware resilience framework that quantifies how heterogeneous users, from highly compliant to consistently reluctant, change a system's ability to resist and contain attack progression. User classes are derived from *observable behavioral telemetry*, such as MFA use, patching, phishing resistance, and backup practices, rather than from subjective labels. The approach combines role-weighted user modeling, safety-practice compliance scoring, residual vulnerability estimation, and a 15-stage four-state Markov formulation (Emergent, Threat, Attack, Compromised). Best Safety Practice (BSP) values are derived from NIST and CISA guidance, while Measured Safety Practice (MSP) values represent observed user or role compliance captured through operational logs. Simulation across five heterogeneous organizational scenarios shows that strong administrator and user compliance maintains the lowest compromise probability, uniformly weak compliance produces the highest, and the ordering of asymmetric cases depends jointly on heterogeneous user postures and the weights representing role-based privilege and operational influence. The framework provides an interpretable decision-support index for comparing user populations, policy configurations, and human-centered defense strategies.

Keywords: Ransomware · Human-centered security · User behavior · Security compliance · Markov chain · Resilience modeling

1 Introduction

Ransomware is a human-targeted cyber threat as much as a technical one. Modern campaigns combine phishing, credential theft, lateral movement, evasion, and backup disruption, allowing attackers to exploit both system weaknesses and human security behavior. As a result, *resilience*, the ability of a system to resist, absorb, and recover from an unfolding attack, depends not only on deployed controls but also on how users and administrators comply with them.

Most ransomware models emphasize malware behavior, vulnerabilities, or technical defenses, while user populations are often treated as homogeneous. Real

organizations, instead, contain highly compliant personnel, moderately trained users, reluctant actors, contractors, remote users, and privileged administrators. These groups differ in authentication behavior, patching responsibility, phishing exposure, backup handling, and incident reporting. Their differences shape the timing and likelihood of ransomware escalation.

This work extends the structural ransomware progression framework in [17] from homogeneous system-level mitigation to heterogeneous user populations. The proposed framework introduces role-weighted MSP aggregation, behavior-conditioned residual vulnerability, and stage-dependent Markov transitions in which user privilege, compliance variability, and uneven control adoption directly influence compromise progression.

The main contributions are: (i) a role-weighted aggregation model that transforms heterogeneous user behavior into system-level MSP; (ii) a residual-risk formulation that combines vulnerabilities, controls, and behavior; (iii) a stage-wise Markov framework for interpretable resilience quantification; and (iv) a comparative scenario analysis showing how user-population structure shifts ransomware resilience across all 15 stages.

Existing ransomware resilience assessments often assume homogeneous users or collapse security posture into a single aggregate score, making it difficult to determine whether resilience is limited by privileged administrators, ordinary users, or heterogeneous behavioral distributions. The proposed framework explicitly models these differences.

The framework provides a quantitative way to study how human behavior shapes ransomware resilience and to support targeted training, privilege redesign, and security investments when calibrated against organizational metrics or public threat intelligence.

The remainder of this paper presents the data-collection challenges, related work, methodology, Markovian risk model, evaluation, discussion, and conclusion.

2 Parameterization Challenges and Calibration Sources

Modeling human-influenced ransomware resilience requires behavioral and system-level parameters that are difficult to observe consistently. Enterprise environments include users with different privileges, tasks, and security habits, while infrastructure spans multiple network zones and platforms with different exposure levels [1, 3, 4, 18]. Mapping vulnerabilities to controls using frameworks such as MITRE ATT&CK, NIST, and CISA guidance requires continuous updates and expert interpretation [12, 19].

The proposed framework separates recommended practice levels from measured practice adherence. Best Safety Practice (BSP) values are derived from established security guidance, including the NIST Cybersecurity Framework, CISA Cross-Sector Cybersecurity Performance Goals, and CISA StopRansomware guidance [20–22]. MSP values measure user, role, or system compliance with those

practices using policy checks, MFA logs, patch-compliance records, phishing simulations, endpoint compliance scores, backup tests, and incident-reporting behavior. Role weights can be estimated from privilege level, system ownership, access to sensitive data or recovery infrastructure, and administrative action frequency. Vulnerability values can be initialized from asset inventories, scanners, CISA KEV, CVE/NVD severity, and ATT&CK mappings [12, 19, 23, 24]. Stage-transition parameters can be calibrated from incident timelines, SOC alerts, red-team traces, tabletop exercises, public ransomware reports, or expert elicitation. Thus, the model is intended for comparative decision support rather than unconditional prediction.

3 Related Works

Ransomware research spans malware analysis, detection, mitigation, threat modeling, and economic analysis [3, 8, 9]. Detection-focused work uses machine learning, anomaly detection, and behavioral signals to identify ransomware before encryption completes [2, 6], while mitigation research studies controls such as segmentation, multi-factor authentication, and SDN-based response [1, 7]. These approaches are valuable for containment but usually do not quantify how heterogeneous users alter compromise progression across a full ransomware kill chain.

Threat-modeling work maps ransomware behavior to structured frameworks such as MITRE ATT&CK [12–14]; techniques such as *Phishing (T1566)*, *Data Encrypted for Impact (T1486)*, and *Inhibit System Recovery (T1490)* correspond directly to ransomware operations. Probabilistic cyber-risk models provide complementary foundations: attack graphs capture intrusion propagation [5], Markov models capture threat evolution [10], Petri nets represent dependencies and concurrency [11], and MDP/POMDP/reinforcement learning approaches model adaptive defense and moving-target strategies [27–29]. This work uses a descriptive Markov formulation because the objective is to quantify resilience under an observed security posture rather than to optimize adaptive defensive actions.

Human-security research motivates the user layer. Ovelgönne et al. analyzed 1.6 million hosts and found that observable behaviors, such as downloaded, unsigned, and low-prevalence binaries, were significantly associated with malware attack volume [25]. Cognitive and simulation studies likewise emphasize that attackers, defenders, and users should not be treated as behaviorally uniform agents [26]. We do not adopt application-use labels such as "gamer" or "software developer"; instead, we rely on the empirical lesson that measurable behavior can define risk-relevant user classes.

The gap is the lack of a compact model that jointly represents user privilege, measured practice adherence, vulnerabilities, network exposure, and stage-wise ransomware progression. This paper addresses that gap with a role-weighted, human-behavior-aware Markov framework for resilience benchmarking and decision support.

4 Proposed Methodology

This work quantifies ransomware resilience by integrating multi-user behavioral modeling, safety-control analysis, vulnerability mitigation, and stage-wise probabilistic attack progression.

4.1 Multi-User Security Posture Estimation

Enterprise users differ in privilege, duties, and adherence to security practices. The framework therefore computes system-level MSP as a role-weighted aggregation of user practices, so privileged accounts exert greater influence. Weights are assigned per system and normalized during aggregation; they reflect operational influence, not security quality, and can be estimated from privilege level, system ownership, access to sensitive data or recovery infrastructure, and administrative action frequency. Compliance classes are modeling abstractions over measured MSP values, not subjective labels or standards-defined categories: NIST and CISA define the BSP targets, while operational telemetry measures MSP. A user-level posture score is computed as a weighted average over role-relevant practices:

$$\text{Posture}[u] = \frac{\sum_p \rho[p]\,\text{MSP}[u,p]}{\sum_p \rho[p]},$$

where $\rho[p]$ denotes the importance of practice p for the role or system context. For scenario construction, high compliance corresponds to $[0.8, 1.0]$, medium to $[0.4, 0.7]$, and low to $[0.0, 0.3]$; a mixed population contains a distribution over these classes. With organizational data, class boundaries can instead be assigned by empirical percentiles.

As a concrete example, an administrator's posture may weight MFA enforcement, privileged credential hygiene, patch approval, backup access control, and incident response more heavily than phishing-reporting behavior. A standard user's posture may instead weight MFA use, phishing resistance, endpoint compliance, password hygiene, and reporting behavior. In a *Good Admin, Poor Users* scenario, administrator MSP values are high for privileged controls while ordinary-user MSP values are low for phishing, credential, or endpoint practices. Role weighting then allows strong administrative hygiene to reduce the risk of later-stage escalation without erasing the initial-access exposure created by weak ordinary-user behavior. Conversely, in a *Poor Admin, Good Users* scenario, compliant ordinary users reduce early exposure, but weak administrative controls still increase risk in privilege escalation, lateral movement, backup neutralization, and impact stages.

The role-weighting mechanism captures the interaction between behavior and privilege. The same MSP weakness has different system-level impact when associated with a standard user, backup operator, system owner, or administrator. For example, weak credential hygiene by a privileged administrator can more strongly influence privilege escalation, backup neutralization, and lateral movement than the same behavior by a low-privilege user. Conversely, strong

standard-user compliance can reduce exposure to phishing and malware delivery without fully compensating for weak administrative hygiene in later ransomware stages.

4.2 Operational Use Workflow

The framework can be applied as a measurement workflow rather than only as a simulation artifact. First, the analyst selects BSP targets from NIST and CISA guidance for ransomware-defense practices. Second, MSP values are collected from available evidence sources, including IAM logs, phishing exercises, endpoint compliance tools, patch management systems, backup tests, and incident response records. Third, user-level posture scores are computed for each role and aggregated using role weights that reflect privilege, access, and operational influence. Fourth, vulnerabilities and safety practices are mapped to ransomware stages. Finally, the Markov model is run under the measured posture or alternative scenarios to identify users, practices, and stages that contribute most to the progression of compromise.

Table 1. Examples of MSP Evidence and Supported Ransomware Stages

Practice	Evidence	Stages
MFA and credentials	MFA use, failed-login patterns, password reuse indicators	Initial Access, Credential Harvesting, Privilege Escalation
Patch/configuration	Patch age, endpoint compliance, secure-baseline drift	Execution, Persistence, Defense Evasion
Phishing/reporting	Click rate, reporting rate, training completion, simulated phishing results	Initial Access, Malware Delivery
Backup/recovery	Restore tests, backup access logs, separation of backup privileges	Backup Discovery/Neutralization, Ransom Launch/Impact
Monitoring/response	EDR/SIEM handling, escalation time, containment records	Defense Evasion, Lateral Movement, Command-and-Control

4.3 Dictionary of Safety Practices

The safety-practice dictionary includes preventive controls such as patching, secure baselines, MFA, and endpoint hardening, as well as reactive controls such as containment, incident response, malware removal, and backup restoration. BSP entries come from NIST and CISA recommendations; when guidance is expressed as ranges or qualitative outcomes, the analyst translates it into an auditable target such as required enrollment, remediation window, restore-test cadence, or logging coverage. MSP entries then measure adherence to those targets for each user, role, or system. Thus, a highly compliant user is high in measured adherence, not "good" in a subjective sense.

4.4 Dictionary of Vulnerabilities

The vulnerability dictionary lists weaknesses commonly exploited by ransomware actors, including unpatched software, exposed services, weak credentials, misconfiguration, insufficient authentication, and inadequate backup or access-control processes. Vulnerabilities are mapped to components, controls, and attack stages.

4.5 Residual Vulnerability and Resilience Quantification

MSP vectors are compared with BSP benchmarks to compute the Resilience Factor (RF), which quantifies the strength of implemented controls relative to recommended practice. A Vulnerability-Control Mapping (VCM) links vulnerabilities to mitigating practices. VCM values can be initialized from ATT&CK-to-control mappings, NIST 800-53 control mappings, scanner findings, and expert assessment; binary values indicate direct mitigation, while fractional values represent partial or indirect mitigation. The computation follows $\mathrm{VCM}, \mathrm{MSP}_{\mathrm{sys}} \rightarrow \mathrm{Mit} \rightarrow \mathrm{CUBE}_{\mathrm{res}} \rightarrow \mathrm{RF}$, connecting control mapping, mitigation strength, residual vulnerability, and stage-wise resilience. Applying this mapping yields a residual vulnerability tensor representing the exploitable surface remaining after mitigation and provides the basis for stage-wise attack progression.

4.6 Petri Nets for Structuring the Ransomware Kill Chain

Petri nets encode dependency, concurrency, and prerequisite relationships in ransomware campaigns. They guide decomposition into fifteen stages: *System Reconnaissance, Initial Access, Malware Delivery, Execution, Persistence Establishment, Privilege Escalation, Defense Evasion, Credential Harvesting, Lateral Movement, Command-and-Control Establishment, Internal Reconnaissance, Data Staging, Backup Discovery/Neutralization, Encryption Deployment, and Ransom Launch/Impact.* Each stage is associated with relevant vulnerabilities and safety practices via the CUBE and VCM tensors, and is modeled as a four-state Markov process whose outcome initializes the next stage.

4.7 Stage-Wise Markov Modeling of Ransomware Progression

The final component translates structural and resilience information into a probabilistic framework using a stage-wise Markov chain. Each of the fifteen stages is represented by four system states: Emergent, Threat, Attack, and Compromised, capturing different levels of adversarial presence within the system. Transition probabilities are derived from resilience factors, residual vulnerabilities, and adversarial capability parameters.

The parameters $\alpha(a)$, $\beta(a)$, and $\delta(a)$ reflect system posture and implemented controls, while $\gamma(a)$ captures escalation from Threat to Attack, $\theta(a)$ governs progression from Attack to Compromised, and $\omega(a)$ represents partial containment that moves the system from Compromised back to Attack without full recovery.

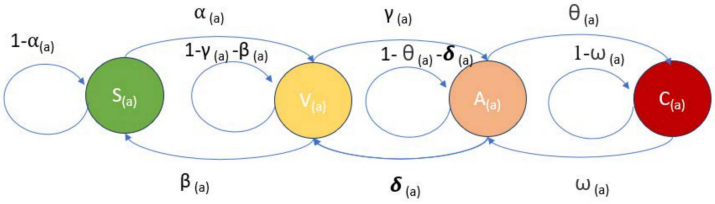

Fig. 1. Stage-wise four-state Markov chain used within each ransomware stage. Cross-stage progression follows the coupling mechanism defined in [17], while the Initial Access gate constrains downstream escalation when the early-stage foothold remains below threshold τ_I.

These parameters define stage-specific transition matrices that govern how the attack evolves across the ransomware kill chain, as illustrated in Fig. 1.

To preserve structural realism, the framework incorporates stage composition and gating across the full 15-stage progression. The probabilistic outcome of each stage is propagated forward to initialize the subsequent stage, ensuring that later stages inherit the security posture established earlier rather than evolving independently. Let a_I denote the Initial Access stage and let $\pi^{(a_I)}$ be its final state distribution. We define the foothold score

$$F_I = \pi^{(a_I)}(S_3) + \pi^{(a_I)}(S_4),$$

and a gate coefficient $g_I = \min(1, F_I/\tau_I)$, where τ_I is the minimum foothold threshold required for unconstrained downstream escalation. For stages $a > a_I$, escalation parameters are scaled as $\gamma_g(a) = g_I\gamma(a)$ and $\theta_g(a) = g_I\theta(a)$. Thus, if early access remains weak ($F_I < \tau_I$), Threat-to-Attack and Attack-to-Compromised transitions are proportionally dampened; if $F_I \geq \tau_I$, downstream progression is unchanged. The threshold τ_I can be calibrated from incident data, exercises, or expert elicitation. This gate prevents unrealistic escalation while keeping the model reproducible.

5 Markovian Model for Risk Analysis

5.1 Multi-User Aggregation of Measured Safety Practices

Organizational environments typically feature multiple users interacting with or jointly operating each system component. These users differ in privilege level, operational roles, and security behavior, and thus contribute differently to the system's effective security posture. To represent this heterogeneity, we define a user-to-system weighting matrix $W[s, u]$ that quantifies the operational or administrative influence of user u on system s.

Let $\mathrm{MSP}[s, u, p] \in [0, 1]$ be user u's measured level for practice p on system s, and let U_s be that system's users. The declared influence weight $W[s, u]$ is

normalized as

$$\lambda_{s,u} = \frac{W[s,u]}{\sum_{j \in U_s} W[s,j]}, \qquad \sum_{u \in U_s} \lambda_{s,u} = 1,$$

where j is only the summation index. For positive MSP values, the system posture is the weighted geometric mean

$$\mathrm{MSP}_{\mathrm{sys}}[s,p] = \exp\left(\sum_{u \in U_s} \lambda_{s,u} \ln\big(\mathrm{MSP}[s,u,p]\big) \right). \tag{1}$$

A zero MSP from any positively weighted user sets the system-level MSP for that practice to zero. Here, s, u, and p index the system, weighted user, and safety practice. This aggregation retains role influence, distinguishes populations with the same weighted arithmetic mean but different user postures, and makes the scalar sensitive to weak-user imbalance; it does not preserve the full distribution after aggregation. The result remains the behavioral input to the downstream resilience and Markov calculations.

The role weights are declared influence assumptions, not coefficients learned by the model. Let G_W denote the weighted geometric MSP, $W_+ = \sum_u W_u$ the total user weight, x_k user k's MSP, and W_k that user's weight. The sensitivity to W_k, with all other values fixed, is

$$\frac{\partial G_W}{\partial W_k} = \frac{G_W}{W_+} \ln\left(\frac{x_k}{G_W} \right).$$

Thus, increasing W_k raises, lowers, or leaves G_W locally unchanged when x_k is above, below, or equal to G_W. For a symmetric illustration, let $x_H > x_L$ be the high and low MSP values and $r = W_A/W_U$ the ratio of total administrator weight W_A to total ordinary-user weight W_U:

$$G_{AP} = (x_H^r x_L)^{1/(r+1)}, \qquad G_{PA} = (x_L^r x_H)^{1/(r+1)}.$$

Here, G_{AP} represents Good Administrator/Poor Users and G_{PA} represents Poor Administrator/Good Users. Hence $G_{AP} > G_{PA}$ for $r > 1$, they are equal at $r = 1$, and the ordering reverses for $r < 1$. This crossover is specific to the symmetric illustration; generally, the aggregate effect depends jointly on heterogeneous user postures and the weights used to represent each user's role-based privilege and operational influence.

5.2 Control-Level and Vulnerability-Aware Resilience Factors

We first define a control-level resilience factor (CRF), denoted $\mathrm{CRF}[s,p]$, representing the contribution of practice p to the resilience of system s based solely on its implementation strength, independent of the vulnerability landscape. Let $\mathrm{MSP}_{\mathrm{sys}}[s,p]$ denote the system-level implementation level of practice p, and

$\mathrm{BSP}[s,p]$ the recommended best-practice level for that system and control. The control-only resilience factor is defined as

$$\mathrm{CRF}[s,p] = 1 - \exp\left(-c\,\frac{\mathrm{MSP}_{\mathrm{sys}}[s,p]}{\mathrm{BSP}[s,p]}\right), \tag{2}$$

where c is a normalizing constant chosen so that full adherence to best practices yields the desired resilience level r. For example, if full implementation corresponds to $r = 0.8$, then $c = -\ln(1-r) \approx 1.609$. When $\mathrm{MSP}_{\mathrm{sys}}[s,p] = \mathrm{BSP}[s,p]$, we obtain $\mathrm{CRF}[s,p] = r$, indicating that the system achieves 80% resilience with 20% residual vulnerability under strict best-practice adherence. This exponential formulation captures diminishing returns as MSP approaches BSP.

Although $\mathrm{CRF}[s,p]$ quantifies the implementation strength of a safety practice, it does not capture the actual vulnerability landscape. Two systems may share similar MSP levels but exhibit very different resilience if exploitable vulnerabilities remain in one but not the other. We therefore extend the control-only measure to a vulnerability-aware resilience factor that combines control strength and residual vulnerability.

Let $\mathrm{CUBE}[a,s,v]$ denote the presence or severity of vulnerability v during attack stage a in system s. Let $\mathrm{VCM}[v,p] \in [0,1]$ quantify how effectively practice p mitigates vulnerability v. For each stage a, system s, and practice p, the pre-mitigation vulnerability load relevant to p is

$$L[a,s,p] = \sum_{v} \mathrm{CUBE}[a,s,v]\,\mathrm{VCM}[v,p].$$

We define a binary relevance indicator

$$K[a,s,p] = \begin{cases} 1, & L[a,s,p] \geq 1, \\ 0, & L[a,s,p] = 0, \end{cases}$$

which indicates whether practice p is relevant in stage a for system s.

The mitigation contribution of control p against vulnerability v is modeled as

$$m[s,v,p] = \mathrm{MSP}_{\mathrm{sys}}[s,p]\,\mathrm{VCM}[v,p].$$

This multiplicative formulation assumes conditional independence among the mitigation contributions of individual safety practices for a given vulnerability. The assumption is applied only at the level of control aggregation and does not imply independence across users, systems, or attack stages. While real-world controls may exhibit correlations or shared dependencies, this approximation provides a tractable formulation that captures diminishing returns when multiple controls are applied to the same vulnerability; future extensions could replace it with dependence-aware aggregation when empirical control-correlation data are available.

$$\mathrm{Mit}[s,v] = 1 - \prod_{p} \big(1 - m[s,v,p]\big).$$

The residual vulnerability in stage a is

$$\mathrm{CUBE}_{\mathrm{res}}[a,s,v] = \mathrm{CUBE}[a,s,v]\,\big(1-\mathrm{Mit}[s,v]\big).$$

The post-mitigation vulnerability load relevant to practice p is then

$$L_{\mathrm{res}}[a,s,p] = \sum_{v} \mathrm{CUBE}_{\mathrm{res}}[a,s,v]\,\mathrm{VCM}[v,p].$$

We define a residual vulnerability ratio

$$\mathrm{VulnRes}[a,s,p] = \begin{cases} \dfrac{L_{\mathrm{res}}[a,s,p]}{L[a,s,p]}, & L[a,s,p] > 0, \\ 0, & L[a,s,p] = 0, \end{cases}$$

where VulnRes $= 1$ indicates that all relevant vulnerabilities remain exploitable, VulnRes $= 0$ indicates complete mitigation, and intermediate values represent partial mitigation.

Residual vulnerability reduces the effective contribution of a control. We therefore define the adequate MSP as

$$\mathrm{MSP}^{\mathrm{eff}}[a,s,p] = \mathrm{MSP}_{\mathrm{sys}}[s,p]\,\big(1-\mathrm{VulnRes}[a,s,p]\big),$$

and the vulnerability-aware resilience factor as

$$\mathrm{RF}[a,s,p] = 1 - \exp\left(-c\,\frac{\mathrm{MSP}^{\mathrm{eff}}[a,s,p]}{\mathrm{BSP}[s,p]}\right).$$

This formulation yields high RF when controls are strong and residual vulnerabilities are small, low RF when vulnerabilities remain exploitable despite high MSP, and stage-specific RF values that reflect the evolving exposure landscape throughout ransomware progression.

Association of Safety Practices and Vulnerabilities The entry point to any attack lies in the weaknesses present in system components. NIST defines a vulnerability as a weakness in computational logic or configuration that, when exploited, undermines confidentiality, integrity, or availability. Mitigation typically involves applying specific safety practices depending on the component and vulnerability type. Software vulnerabilities may require code changes, protocol hardening, or deprecation, while physical vulnerabilities may require operational checks such as lock verification or device access controls.

Each vulnerability class v is associated with one or more safety practices p. The matrix $\mathrm{VCM}[v,p]$ encodes these relationships, with $\mathrm{VCM}[v,p] = 1$ indicating that practice p mitigates vulnerability v, and $\mathrm{VCM}[v,p] = 0$ indicating no direct mitigation relationship. At the attack-stage level, the three-dimensional CUBE tensor $\mathrm{CUBE}[a,s,v]$ identifies which vulnerabilities in which systems are involved in each attack stage. A successful multi-stage ransomware campaign

typically requires multiple such sub-attacks to succeed across different systems and vulnerabilities.

Multiplying CUBE$[a, s, v]$ by VCM$[v, p]$ yields $L[a, s, p]$, which indicates how a given practice, applied to a system, mitigates vulnerabilities during stage a. This is binarized into $K[a, s, p]$ as described above and used to activate the relevant controls during RF, β, and δ computations.

Preventive safety practices—such as patching, configuration hardening, and strong access control—primarily reduce the likelihood of transitions from safer states to more exposed states (e.g., Emergent $\rightarrow$ Threat). Reactive practices, such as incident response, restoration, and containment, primarily increase the probability of recovery transitions from more severe to less severe states (e.g., Attack $\rightarrow$ Threat or Compromised $\rightarrow$ Attack). Together, these practices shape the Markov model's transition parameters.

5.3 Network-Level Propagation

Ransomware propagation is inherently network-driven, but this work uses a simplified zone-based abstraction rather than explicit node-to-node propagation. Each system s is assigned to a logical zone $z(s)$ (e.g., internet-facing, internal, restricted), and each zone is associated with an exposure weight $Z[z(s)] \in [0, 1]$ representing its relative attacker reachability.

This exposure term is used only as a modest modifier of the Emergent $\rightarrow$ Threat transition. Thus, for stage a and system s, the base probability $\alpha(a, s)$ is adjusted by zone exposure so that more exposed zones have slightly higher entry risk than restricted ones. This preserves residual vulnerability as the dominant driver while allowing network placement to influence progression.

Explicit network exposure matrices, node-level lateral movement, and dynamic inter-system coupling are left for future work. The present abstraction keeps parameter estimation tractable while still capturing first-order segmentation effects.

5.4 Markov State Transitions and Parameters

Each attack stage uses four states: Emergent (S_1), Threat (S_2), Attack (S_3), and Compromised (S_4). The stage-wise construction and parameter derivation follow our earlier model [17]; the present extension conditions them on the multi-user MSP and residual-vulnerability measures developed above.

The transition matrix for a given stage has the general structure

$$T^{(a)} = \begin{bmatrix} 1-\alpha(a) & \alpha(a) & 0 & 0 \\ \beta(a) & 1-\beta(a)-\gamma(a) & \gamma(a) & 0 \\ 0 & \delta(a) & 1-\delta(a)-\theta(a) & \theta(a) \\ 0 & 0 & \omega(a) & 1-\omega(a) \end{bmatrix}$$

Here, α, γ, and θ govern forward progression, while β, δ, and ω represent regression, containment, or recovery. Parameters lie in $[0, 1]$, with $\beta + \gamma \leq 1$ and

$\delta + \theta \leq 1$ so that $T^{(a)}$ is row-stochastic. In the reported experiment $\omega = 0$, because compromised-system isolation and recovery are outside scope. The remaining terms are obtained from stage-relevant resilience, vulnerability, and attacker-pressure inputs; full derivations are given in [17].

The Markov formulation assumes that, conditional on the current state and attack stage, the next-state distribution does not depend on earlier history; transition probabilities are also treated as constant within each simulated stage and the four states as adequate at the chosen time resolution. These are modeling assumptions and are not empirically validated by the present scenario-based simulations. With longitudinal incident or emulation traces, they should be tested by comparing first- and higher-order transition models, checking transition stability across time windows, and evaluating state-duration fit. If memory, nonstationarity, or unobserved state variation is material, a semi-Markov, time-varying Markov, or hidden Markov model is more appropriate.

6 Evaluation and Case Analysis

This section evaluates the framework through internal validity checks, evidence-grounded parameterization, and scenario-based case analysis. The framework supports comparative scenario analysis rather than absolute breach prediction. We simulate five behavioral scenarios with normalized MSP values in $[0, 1]$, role weights encoding privilege asymmetry, and common attacker-side parameters for $\gamma(a)$, $\theta(a)$, and $\omega(a)$. Every scenario uses the same 27 user records across 12 systems: 11 administrator records and 16 ordinary-user records. User identities, systems, practices, and role weights remain fixed; only MSP values vary. This isolates behavioral effects from population or attack-strength differences. Table 2 summarizes the scenario design used to generate the reported comparisons.

6.1 Evaluation Scope and Evidence Sources

The scenario analysis contrasts the proposed role-weighted representation with two common assumptions: homogeneous posture, which collapses all users into one MSP value, and equal weighting, which treats all users as equally influential. Public sources ground different model layers: NIST, CISA, MITRE, KEV, and CVE/NVD define controls, ransomware stages, and vulnerability mappings [12, 19–24], while threat-intelligence reports support the prevalence of phishing, credential abuse, lateral movement, backup disruption, and recovery challenges [3, 15, 16]. In deployment, MSP and transitions can be calibrated using phishing simulations, IAM/MFA logs, patch reports, SIEM/EDR alerts, and backup recovery logs.

6.2 Behavioral Posture Profiles from Prior Malware Telemetry

Ovelgönne et al. [25] provide large-scale empirical evidence that observable behavior is associated with malware exposure. Their telemetry from approximately

Table 2. Simulation Design Parameters for Scenario Comparison

Element	Setting Used in Reported Scenarios
Compliance classes	High $[0.8, 1.0]$, Medium $[0.4, 0.7]$, Low $[0.0, 0.3]$; Mixed combines values from the three classes.
Population	The same 27 records (11 administrators and 16 ordinary users) across 12 systems; identities, systems, practices, and role weights are fixed.
Role weighting	Role weights encode privilege asymmetry: administrators, backup operators, and system owners receive greater operational influence than standard users.
Attack stages	fifteen ransomware stages from reconnaissance through ransom launch/impact, with the same stage ordering across all scenarios.
Attacker-side terms	$\gamma(a)$, $\theta(a)$, and $\omega(a)$ use common stage templates across scenarios so that differences are driven by MSP composition and role weighting.
Initial Access gate	Downstream escalation are scaled by $g_I = \min(1, F_I/\tau_I)$ after the Initial Access stage.
Comparison horizon	Fig. 4 reports the average compromised-state probability over 100 simulation steps.

1.6 million hosts identifies significant differences in malware attack volume across behavior-derived profiles: activity involving unsigned, low-prevalence, or unique binaries is associated with greater exposure, whereas signed, high-prevalence activity is associated with lower exposure. Our results are qualitatively consistent: stronger measured safety-practice compliance produces lower modeled compromise progression, while broadly weak compliance produces higher progression. Both studies therefore show that treating users as behaviorally homogeneous can conceal risk-relevant variation.

The comparison is methodological and directional, not numerical. Ovelgönne et al. derive profiles from software-exposure telemetry and evaluate observed malware attack volume; our framework derives MSP from NIST/CISA-aligned practices, represents role-based privilege and operational influence through weights, and estimates stage-wise ransomware $P(S_4)$. Their findings support behavior-derived user profiling but do not validate our MSP values, weights, transitions, scenario ordering, or probabilities. Direct validation requires organizational telemetry linking compliance and privilege to observed ransomware-stage outcomes; IAM/MFA records, phishing simulations, patch and endpoint reports, backup tests, and incident records could provide such evidence.

6.3 Simulation Results Across Behavioral Scenarios

Using the simulation design in Table 2, we compare the five administrator–user compliance scenarios summarized in Table 3.

Figure 2 shows the four-state distribution for *Good Admin, Good Users*. At the final Ransom Impact stage, the compromised-state probability is 20.98%.

At the baseline setting ($\gamma = 1.00, T = 100, \tau_I = 0.30, \omega = 0$), the mean final $P(S_4)$ values across the 15 stages are 0.1815 for *Good Admin, Good Users*, 0.8153

Table 3. Behavioral Scenarios Used in the Simulation Study

Scenario	Admin Role	User Role	Interpretation
Good Admin, Poor Users	High	Low	Strong administrative discipline but weak end-user practices.
Poor Admin, Good Users	Low	High	Administrative weaknesses offset by strong user compliance.
Good Admin, Good Users	High	High	Well-defended environment with consistent security practices.
Mixed User Behavior	Medium	Mixed	Heterogeneous user practices with moderate oversight.
Poor Admin, Poor Users	Low	Low	Weak overall security posture and elevated systemic risk.

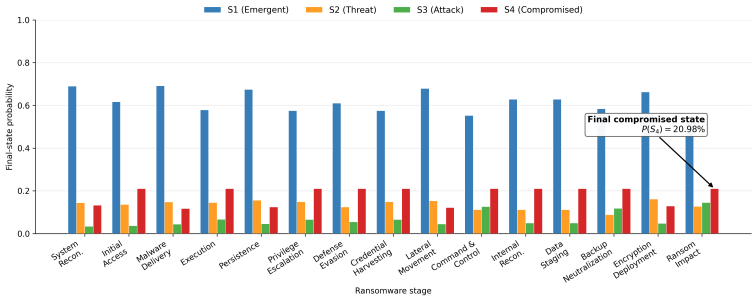

Fig. 2. Final state probabilities per ransomware stage for Good Admin, Good Users.

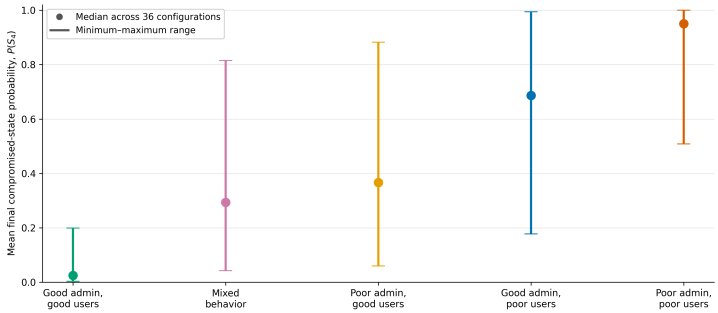

Fig. 3. Targeted sensitivity of the stage-averaged $P(S_4)$ at the final simulation time. Points denote medians and whiskers denote minimum–maximum ranges over the tested parameter grid.

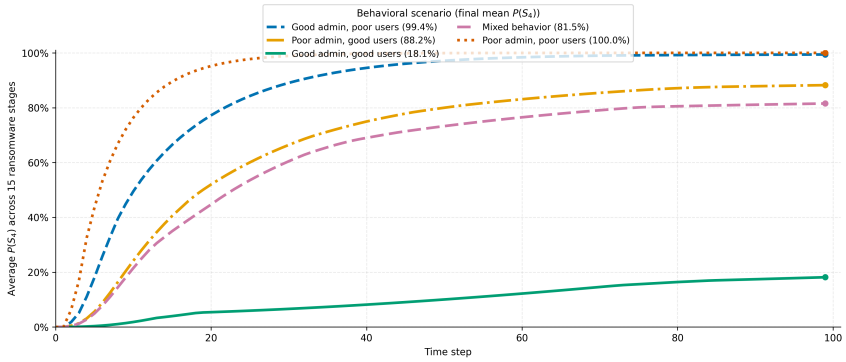

Fig. 4. Average S_4 (Compromised state) probability over time across all five behavioral scenarios. Percentages in the legend report the final time-step mean across the 15 stages.

for *Mixed User Behavior*, 0.8823 for *Poor Admin, Good Users*, 0.9941 for *Good Admin, Poor Users*, and 1.0000 (rounded) for *Poor Admin, Poor Users*. Figure 4 shows their trajectories. These results show that compromise progression depends jointly on heterogeneous user postures and the role-based privilege and operational influence represented by the weights; neither posture nor privilege alone determines the scenario ordering.

6.4 Targeted Parameter Sensitivity

We tested all 36 combinations of $\gamma \in \{0.4, 0.6, 0.8, 1.0\}$, horizon $T \in \{25, 50, 100\}$, and Initial Access threshold $\tau_I \in \{0.2, 0.3, 0.4\}$, with $\omega = 0$. Figure 3 reports the minimum, median, and maximum mean final $P(S_4)$ for each scenario. The ordering remained Good Admin/Good Users, Mixed Behavior, Poor Admin/Good Users, Good Admin/Poor Users, and Poor Admin/Poor Users in all 36 configurations. This targeted sweep is not a global uncertainty analysis over all model inputs.

7 Discussion

Under the reported assumptions, jointly strong behavior yields the lowest compromise and jointly weak behavior the highest; the scenario ordering remained stable over the tested transition grid. For asymmetric scenarios, the analytical crossover shows that the aggregate effect is determined jointly by measured user postures and by the weights encoding role-based privilege and operational influence. Credential extraction and lateral movement remain important escalation points, consistent with reports emphasizing credential abuse, privilege misuse, lateral propagation, backup disruption, and recovery challenges [3, 15, 16]. The Initial Access gate constrains infeasible downstream escalation. Lateral movement is represented as a stage-level transition effect rather than explicit

graph traversal, an abstraction for tractable posture comparison rather than a topology-aware propagation model.

The evaluation demonstrates comparative behavior under controlled assumptions; it does not establish calibrated breach prediction. Operational accuracy requires organization- or incident-derived MSP, vulnerability, and transition estimates. Validation would require longitudinal evidence combining user compliance, privilege, attack progression, and recovery outcomes, using sources such as phishing exercises, IAM/MFA and patch records, EDR/SIEM telemetry, and backup tests. The framework can then be recalibrated for ransomware or other multi-stage attacks by replacing the stage decomposition and estimating the corresponding transitions.

8 Conclusion

This paper presented a human-behavior-aware ransomware resilience framework combining role-weighted users, residual vulnerability, Petri-net kill-chain structure, and stage-wise Markov progression. Unlike homogeneous-posture models, it represents how administrators and ordinary users differently affect attack progression through measured safety-practice compliance.

Internal checks confirm row-stochastic transitions, bounded outputs, and effective Initial Access gating, but do not empirically establish memorylessness. Across the tested transition grid, jointly strong behavior consistently yields the lowest compromise, jointly weak behavior the highest, and the reported scenario ordering remains stable. More generally, asymmetric and mixed outcomes depend jointly on heterogeneous user postures and the weights representing role-based privilege and operational influence. The stage patterns remain consistent with ransomware reports on credential abuse, privilege misuse, lateral movement, backup disruption, and recovery challenges. The framework supports comparison of organizational human-security postures under stated assumptions.

References

1. Akbanov, M., Vassilakis, V.G., Logothetis, M.D.: WannaCry ransomware: Analysis of infection, persistence, recovery prevention and propagation mechanisms. J. Telecommun. Inf. Technol. 1, 113–124 (2019)
2. Kok, S.H., Abdullah, A., Jhanjhi, N.Z., Supramaniam, M.: Prevention of crypto-ransomware using a pre-encryption detection algorithm. Computers 8(4), 79 (2019)
3. Razaulla, S. et al.: The age of ransomware: Evolution, taxonomy, and research directions. IEEE Access 11, 40698–40723 (2023)
4. Chandramouli, R., Pinhas, D.: Security Guidelines for Storage Infrastructure. NIST Special Publication 800-209 (2020). DOI: 10.6028/NIST.SP.800-209
5. Abraham, S., Nair, S.: Cyber security analytics: A stochastic model for security quantification using absorbing Markov chains. J. Commun. 9(12), 899–907 (2014)
6. Al-rimy, B.A.S., Maarof, M.A., Shaid, S.Z.M.: Crypto-ransomware early detection model using novel incremental bagging with enhanced semi-random subspace selection. Future Gener. Comput. Syst. 101, 476–491 (2019)
7. Keshavarzi, M., Ghaffary, H.R.: I2CE3: A dedicated and separated attack chain for ransomware offenses as the most infamous cyber extortion. Comput. Sci. Rev. 36, 100233 (2020)
8. Kharraz, A., Robertson, W., Balzarotti, D., Bilge, L., Kirda, E.: Cutting the Gordian knot: A look under the hood of ransomware attacks. In: Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA 2015), vol. 12, pp. 3–24. Springer (2015)
9. Conti, M., Gangwal, A., Ruj, S.: On the economic significance of ransomware campaigns: A Bitcoin transactions perspective. Comput. Secur. 79, 162–189 (2018)
10. Gore, R., Padilla, J., Diallo, S.: Markov chain modeling of cyber threats. J. Defense Model. Simul. 14(3), 233–244 (2017)
11. Yu, W., Ding, Z., Liu, L., Wang, X., Crossley, R.D.: Petri net-based methods for analyzing structural security in e-commerce business processes. Future Gener. Comput. Syst. 109, 611–620 (2020)
12. MITRE Corporation: MITRE ATT&CK®: A knowledge base of adversary tactics and techniques (2023). `https://attack.mitre.org`, accessed 8 July 2026
13. MITRE Corporation: ATT&CK for ransomware: Mapping malicious behavior to ATT&CK (2021). `https://www.mitre.org/publications/technical-papers/attck-for-ransomware`, accessed 8 July 2026
14. Zand, A., Alshamrani, A., Sitnikova, E.: Using MITRE ATT&CK framework to detect ransomware attacks. In: Proc. IEEE Int. Conf. Intell. Secur. Informatics (ISI), pp. 1–6 (2021)
15. Microsoft Threat Intelligence: Ransomware as a service: Understanding the cybercrime gig economy and how to protect yourself (2022). Microsoft Security Blog, accessed 8 July 2026
16. CISA, FBI, MS-ISAC: #StopRansomware: LockBit 3.0 Ransomware. Cybersecurity Advisory AA23-165A (2023). `https://www.cisa.gov/news-events/cybersecurity-advisories/aa23-165a`, accessed 8 July 2026
17. Kembamba, F., Khan, J.I.: Cyberinfrastructure Resilience Index against Ransomware Using Markov Chain Modeling and Tensor Computation. In: Proceedings of the 12th International Conference on Information Systems Security and Privacy (ICISSP), vol. 2, pp. 189–196 (2026)
18. National Institute of Standards and Technology: Energy Sector Asset Management. NIST Special Publication 1800-23 (2020). DOI: 10.6028/NIST.SP.1800-23

19. MITRE Center for Threat-Informed Defense: NIST 800-53 Controls to ATT&CK Mappings (2023). https://center-for-threat-informed-defense.github.io/mappings-explorer/, accessed 8 July 2026
20. National Institute of Standards and Technology: The NIST Cybersecurity Framework (CSF) 2.0. NIST Cybersecurity White Paper (2024). DOI: 10.6028/NIST.CSWP.29
21. Cybersecurity and Infrastructure Security Agency: Cross-Sector Cybersecurity Performance Goals (CPGs) (2023). https://www.cisa.gov/cross-sector-cybersecurity-performance-goals, accessed 8 July 2026
22. Cybersecurity and Infrastructure Security Agency: StopRansomware Guide (2023). https://www.cisa.gov/stopransomware/ransomware-guide, accessed 8 July 2026
23. Cybersecurity and Infrastructure Security Agency: Known Exploited Vulnerabilities Catalog. https://www.cisa.gov/known-exploited-vulnerabilities-catalog, accessed 8 July 2026
24. National Institute of Standards and Technology: National Vulnerability Database. https://nvd.nist.gov, accessed 8 July 2026
25. Ovelgönne, M., Dumitras, T., Prakash, B.A., Subrahmanian, V.S., Wang, B.: Understanding the relationship between human behavior and susceptibility to cyber attacks: A data-driven approach. ACM Trans. Intell. Syst. Technol. 8(4), Article 51 (2017)
26. Veksler, V.D., Buchler, N., Hoffman, B.E., Cassenti, D.N., Sample, C., Sugrim, S.: Simulations in cyber-security: A review of cognitive modeling of network attackers, defenders, and users. Front. Psychol. 9, 691 (2018)
27. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley, New York (1994)
28. Zheng, J., Namin, A.S.: Markov decision process to enforce moving target defence policies. arXiv:1905.09222 (2019)
29. Dutta, A., Chatterjee, S., Bhattacharya, A., Halappanavar, M.: Deep reinforcement learning for cyber system defense under dynamic adversarial uncertainties. arXiv:2302.01595 (2023)