**Evaluating a New Threat Modeling Protocol in Horizontal Organizations**

**Evaluation Objectives and Key Metrics**

Evaluating the new threat modeling protocol involves comparing it against the well-known STRIDE framework in real-world horizontal (non-hierarchical) organizations. The goal is to assess how well the new protocol identifies and mitigates digital security threats, physical security risks, and operational security issues, all while fitting the organization’s decentralized governance style. **Multiple metrics** are used to ensure a comprehensive evaluation (template.pdf) (template.pdf). As highlighted in the literature, objective criteria are critical for comparing the effectiveness and context-fit of threat modeling frameworks (template.pdf). The evaluation will focus on the following key metrics:

* **Usability** – Measures how easy and clear the protocol is for participants to use in practice. This includes the clarity of the process steps and the effort required to learn and apply the method. *Rationale:* In horizontal teams, a complex or cumbersome process can deter participation. Prior research notes that frameworks integrating democratic processes can face usability challenges (e.g. a steep learning curve or “voting fatigue” in large groups) (template.pdf) (template.pdf). **Data collection:** After each case study session, collect qualitative feedback from users on the protocol’s clarity, ease-of-use, and any difficulties encountered (e.g. via questionnaires or brief interviews) (template.pdf). High usability would be reflected by positive participant feedback and minimal confusion during the modeling exercise.
* **Precision** – The accuracy of threat identification. This metric looks at how many of the threats identified by the protocol are truly relevant and valid (true positives) versus false alarms. **Definition:** the number of correctly identified threats relative to the total threats identified (template.pdf). *Rationale:* A precise threat model minimizes false positives, focusing effort on real issues. **Data collection:** For each case study, have security experts compile a reference set of actual threats present in the scenario (or use the union of threats identified across methods as a superset). Compare the protocol’s identified threats to this reference. Compute precision as Correctly Identified Threats ÷ Total Threats Identified (template.pdf). This requires judging which identified threats are valid; this judgment can come from expert review or consensus among participants after the session.
* **Recall** – The coverage of threat identification. This measures how many of all *actual* threats the protocol managed to catch. **Definition:** the proportion of existing threats (in the reference model) that were identified by the protocol (template.pdf). *Rationale:* A high recall means the protocol is not missing critical threats (minimizing false negatives). **Data collection:** Using the same reference threat set, compute recall as Correctly Identified Threats ÷ Total Actual Threats (template.pdf). For example, if the reference list has 20 potential threats and the protocol found 15 of them, recall = 15/20 = 75%. Ensuring a broad threat coverage is especially important when dealing with diverse threat types (cyber, physical, operational).
* **Operational Efficiency** – Assesses the time and effort required to perform threat modeling and implement mitigations. In this context, one concrete measure is the **latency** or total time to complete each phase of the threat modeling process (template.pdf). *Rationale:* Horizontal organizations often have limited formal structure, so efficient processes are needed to avoid meeting fatigue and delay. **Data collection:** Track the duration of each threat modeling session (from start to finish, and possibly time per step if granular data is needed) and the person-hours involved. Also record the **response time** and workload when reacting to identified issues or simulated incidents. Lower time and smoother execution indicate higher operational efficiency. Additionally, note any bottlenecks (e.g. lengthy deliberations on decisions) as part of qualitative observations.
* **Adaptability** – How easily the protocol adapts to changes in the environment or scale of the organization. This includes adapting to new threat information, changes in team structure, or different contexts (digital vs. physical scenarios). The literature emphasizes that adaptability to changing operational environments is a fundamental criterion (template.pdf). *Rationale:* Decentralized groups and their systems evolve quickly; the threat model should remain useful as conditions change. **Data collection:** Throughout the case studies, introduce changes or new information – for example, a new technology introduced mid-project or a shift in the organization’s structure – and observe if the threat model can be updated without starting from scratch. In the long-term study, track how the protocol handles real changes in the organization over time (e.g. growth in membership, new assets) (template.pdf). An adaptable protocol will incorporate changes with minimal friction, maintaining effectiveness.
* **Scalability** – The protocol’s ability to function in larger groups and more complex systems without degradation of performance. This looks at whether the method can handle many participants or a broad scope of assets and still produce useful results. Prior work notes that scalability and ability to handle complex organizational structures are critical factors for modern threat modeling frameworks (template.pdf). *Rationale:* Horizontal organizations can range from small collectives to large global networks; the protocol must work for both small and large scales. **Data collection:** Apply the protocol in organizations of different sizes (a small cooperative vs. a large distributed community) and compare outcomes. Monitor metrics like the number of contributors actively involved, and whether the process still concludes in reasonable time as scale increases. Also, evaluate if the threat list or model remains manageable or if it grows too unwieldy as scope increases. Techniques from literature (e.g. scenario fusion to reduce complexity (template.pdf)) can be noted if used by the protocol to remain scalable.
* **Preservation of “Democratic Centralism” (Horizontality)** – This metric gauges how well the protocol preserves the organization’s horizontal character and decision-making style. In practical terms, does the threat modeling process allow broad, democratic participation and collective deliberation, and does it avoid introducing hierarchy or power imbalances? *Rationale:* A key design goal is that the security process itself should not undermine the flat governance structure. The literature review stresses that frameworks for horizontal contexts must respect and strengthen distributed governance, equitable participation, and the absence of formal centralization (template.pdf). In other words, the protocol should harmonize collective input with coordinated action – an idea rooted in **democratic centralism**, where group deliberation is balanced with unified execution of decisions (template.pdf). **Data collection:** Qualitative assessments are primary here. Observe decision-making during threat modeling sessions: are decisions about threats and mitigations made by consensus or at least majority vote? Are all voices heard, or do a few experts dominate? After sessions, interview participants about their perceptions: Did they feel the process was collaborative and in line with their democratic practices? Did the protocol improve transparency of security decisions? Also, examine if the group had to create any temporary “centralized” roles (e.g. appointing a security leader) to use the protocol and whether that was done in a way consistent with their norms (for example, by group mandate and with accountability). Ideally, the protocol should enable **collective decision-making with cohesive implementation**, mirroring the principle that even in a flat group, once a decision is democratically made, it is executed in a coordinated way (template.pdf). A positive outcome would be that participants report the security modeling felt inclusive, and any needed leadership (for efficiency) was temporary and agreed-upon, not a permanent shift in power.

Each metric provides a different lens on the protocol’s effectiveness. **Quantitative metrics** like precision, recall, time, and participation counts will be collected for objective comparison. **Qualitative insights** on usability and governance fit are gathered to ensure the protocol aligns with the social context of horizontal teams. This multi-metric approach is grounded in best practices from prior frameworks analysis, where effectiveness, adaptability, and integration of social factors are all considered in judging a security framework (template.pdf) (template.pdf).

**Case Study Evaluation Methodology**

To evaluate the new threat modeling protocol in practice, a series of case studies will be conducted in real organizations. The evaluation is designed as an **experimental, comparative study** involving groups with different degrees of horizontality (template.pdf). In each case study, the new protocol is applied and compared side-by-side with STRIDE as a baseline. Using multiple organizations (e.g. a tech worker cooperative, a decentralized open-source project, and a traditional company for contrast) ensures the protocol is tested in varied real-world settings. All participants will receive training to ensure a fair comparison between methods. The methodology consists of three main phases:

1. **Parallel Threat Modeling Sessions (Protocol vs. STRIDE):** For each organization selected, conduct two threat modeling sessions in parallel on the *same* security scenario. One group of participants uses the **new protocol**, and another group (from the same organization) uses **STRIDE** to model threats for an identical system or process (template.pdf). *Setup:* Before the sessions, provide structured training workshops on both the new protocol and STRIDE (template.pdf) so that participants are familiar with each approach. This ensures that any differences observed are due to the frameworks themselves rather than lack of understanding. *Execution:* The two groups (which ideally have similar composition in terms of skill mix) perform threat modeling concurrently or back-to-back. Both sessions are observed by researchers who document the process steps, group interactions, questions asked, and any difficulties. Time stamps are recorded for each phase of the process (e.g. diagramming the system, identifying threats, prioritizing mitigations). *Data collection:* After the exercise, collect the outputs – the list of identified threats and proposed mitigations from each group. These outputs will be analyzed with the metrics defined above (precision, recall, etc.). For example, to evaluate **precision and recall**, compare the threats identified by each group to a reference threat set compiled by an expert panel (template.pdf). To evaluate **operational efficiency**, note the total time each group took and how that scales with the number of participants (template.pdf). *Post-session:* Administer a survey or hold a debrief with participants to capture **usability feedback** – what did they think of the method they used? Was anything confusing or particularly helpful? (template.pdf) This provides qualitative data on user satisfaction and perceived clarity of each approach.

Throughout all phases, maintaining **ethical and participatory research practices** is important, especially in non-hierarchical organizations. Researchers should obtain group consent, be transparent about the goals of the study, and involve the community in interpreting results whenever possible. This approach not only yields better data (because participants trust the process and engage fully) but also mirrors the ethos of horizontality in the evaluation itself.

**Challenges in Evaluating Security Frameworks for Decentralized Organizations**

Assessing a security framework in a decentralized, horizontal organization comes with unique challenges. Traditional evaluation methods or metrics may not directly apply, and the organizational culture can affect how the evaluation must be conducted. Key challenges include:

* **Cultural and Structural Misalignment:** Many security methodologies (like classical access control or threat modeling tools) implicitly assume a hierarchical structure (with clear roles like admins, managers, etc.). In a horizontal group, such assumptions break down. For instance, conventional role-based access control can conflict with collectively agreed policies, forcing a hierarchy where none exists (template.pdf). During evaluation, this means a new protocol might need to **fit into a flat governance model**, which is harder to measure by standard means. A related challenge is **decision-making speed**: without a top-down chain of command, reaching agreement on security decisions (e.g. how to respond to a threat) can take longer, affecting metrics like operational efficiency. Evaluators must account for the fact that a slower decision process might be a deliberate trade-off for inclusivity, rather than a flaw.
* **Unique Threats in Horizontal Environments:** Decentralized organizations face certain threats that are rare in hierarchical settings, making it challenging to establish evaluation benchmarks. For example, **Sybil attacks** (fake identities to sway group decisions) or **quorum manipulation** (exploiting the rules of consensus) target the governance mechanism itself (template.pdf). Traditional threat models like STRIDE don’t explicitly cover these socio-technical attacks. Therefore, evaluating the new protocol’s recall/precision on such threats requires the researchers to define these threats clearly and perhaps simulate them (as we do in the methodology). Another example is the **“digital vanguard”** phenomenon: individuals who control critical digital resources (admin accounts, communication channels) might accumulate disproportionate power in a flat group (template.pdf). A threat modeling protocol might flag this as an operational security risk (insider threat or single point of failure), but measuring improvement is tricky – it could involve qualitative change (e.g. the organization rotates credentials more often). The challenge for evaluators is to capture whether the protocol truly mitigates these unique threats without established quantitative baselines.
* **Participant Engagement and Fatigue:** Horizontal organizations rely on volunteer or member participation. Conducting lengthy security exercises or frequent threat modeling meetings could lead to **participation fatigue**. As noted in the literature, requiring frequent decision votes (for example, approving access each time via a democratic process) can exhaust members in large groups (template.pdf). During evaluation, if participants become disengaged or drop out of the process, it skews results – a protocol might look unusable not because it’s inherently flawed, but because the evaluation asked too much of busy volunteers. Ensuring genuine engagement in case studies is challenging; researchers must schedule activities in a way that respects participants’ time and energy. Moreover, interpreting usability feedback can be nuanced – critical comments might reflect the **learning curve** (which could be overcome with more practice) rather than an inherent problem, especially if people are not accustomed to structured security planning. We must distinguish between initial friction and long-term usability issues.
* **Measuring Intangible Outcomes:** Some goals of a security protocol in a horizontal org are qualitative, like **empowering members** or **improving trust** in the system. These don’t lend themselves to straightforward metrics. While we have proxies (e.g. survey results, participation rates, or the “preservation of democratic centralism” metric discussed), it remains a challenge to convincingly quantify concepts like “did this protocol maintain our democratic ethos?” There’s also the issue of **demonstrating prevention** – if the protocol is effective, certain attacks won’t happen. How do we give credit for an attack *not* happening? This often requires historical comparison or control groups, which in the real world of unique organizations is hard to establish. Additionally, horizontal organizations might have an inherent skepticism of formal metrics or outsider evaluation, seeing them as potentially imposing hierarchy. As a result, evaluators must be sensitive and perhaps lean more on collaborative assessment (e.g. co-interpreting results with members) rather than just issuing a score.
* **Logistical and Ethical Constraints:** Unlike a controlled corporate environment, decentralized groups may have irregular schedules, diverse geographical distribution, or consensus-required decision-making even to participate in a study. Coordinating the evaluation (setting up sessions, simulations, etc.) might require lengthy discussions and approvals by the collective, which can delay the process. Ethically, the evaluator must ensure transparency and that the research itself doesn’t violate the group’s principles. For example, running a surprise security drill (like a simulated phishing attack) typically requires buy-in so that it’s not seen as a breach of trust by the researchers. Balancing the need for realistic testing with informed consent is a delicate challenge in these communities.

Despite these challenges, addressing them head-on in the research design (as described in the methodology) helps ensure the evaluation is fair and the insights are valid for the context of horizontal organizations. Flexibility and open communication with participant organizations are crucial to overcome these hurdles.

**Best Practices for Assessment in Horizontal Contexts**

Based on lessons from prior studies and the unique needs of non-hierarchical groups, here are best practices to effectively evaluate security frameworks like the new threat modeling protocol in a decentralized setting:

* **Collaborative Planning and Training:** Work *with* the organization to plan the case study. Rather than imposing an evaluation plan, co-design it in a way that aligns with their workflow. For example, integrate threat modeling sessions into their regular meetings or decision processes so it feels natural. Provide thorough training and education sessions in advance (template.pdf), emphasizing how the protocol supports the group’s values (e.g. “this will help us protect our project *together*”). When people see the relevance and understand the tool, they are more likely to participate earnestly. Additionally, encourage the organization to designate a small internal facilitation team (not a top-down leader, but maybe a rotating role or a subgroup) to champion the process – this respects their autonomy while ensuring someone is looking after the evaluation tasks from their side.
* **Use Baseline Comparisons to Highlight Value:** Including a comparison to STRIDE (or another well-known framework) is valuable not just for research rigor but for the participants’ insight. It helps to show concrete differences – for instance, participants might observe that STRIDE missed an organizational process issue that the new protocol caught, illustrating its value. From a research perspective, having a baseline group in the same organization acts as a control for context factors (template.pdf). Make sure both groups (protocol and STRIDE) share their experiences in a debrief; this can even become a learning exercise for the org. Just be cautious to avoid a competitive tone – the aim is to learn which aspects work best in horizontality. If one method clearly identifies more physical threats, for example, that’s an insight for everyone.
* **Multi-Dimensional Metrics and Data:** Combine **quantitative and qualitative data** for a holistic evaluation (template.pdf) (template.pdf). In practice, this means not relying on just one score or number. Use the precision/recall/time measurements to get objective evidence of performance, but also heavily weigh participant feedback and observations. A method that scores slightly lower on recall but is loved by users might be more fitting in the long run for a horizontal group, because adoption and consistent use are crucial. Conversely, if a method finds many threats but people hate using it, it will likely be abandoned in practice – a point that raw metrics would miss. Conducting follow-up interviews after each phase can unveil *why* certain metrics turned out as they did, offering guidance on how to improve the protocol. For instance, if the new protocol took longer than STRIDE in the session, was it because of more discussion (which could actually mean more learning) or because the steps were unclear? Such nuance comes from qualitative insights.
* **Realistic Scenario Testing:** Incorporate scenarios that reflect the *actual threat landscape* for decentralized orgs, including those hybrid socio-technical threats (template.pdf). This best practice is evident in our second phase where we simulate attacks like Sybil attacks, quorum manipulation, as well as more conventional cyber or physical breaches. By doing so, we ensure the evaluation covers not only classic IT security issues (e.g. malware, data breach) but also the process-oriented threats that can undermine a collective. It’s important to tailor these scenarios carefully: they should be challenging but not catastrophic, and relevant to the organization’s activities. Afterwards, debrief with the group to discuss what happened – this turns the evaluation into a learning opportunity, reinforcing security awareness. It’s a best practice to let the group self-reflect: *did our current practices handle this? what could we do better?* Their reflections can be recorded as additional data (and also serve as a secondary validation of whether the threats identified in modeling were on-point). Simulated drills should be used sparingly (to avoid fatigue or panic), but when done with consent, they provide high-value insight into how theory meets practice.
* **Iterative and Phased Evaluation:** Conduct the evaluation in **phases**, as we have planned (workshop, simulation, long-term study), instead of a one-off test (template.pdf) (template.pdf). This phased approach is a best practice because it mirrors how adoption of a security protocol would naturally roll out. After each phase, take time to analyze results and, if possible, refine the protocol or how it is used before moving to the next phase. For example, if after the parallel sessions some usability issues are found, the protocol guide can be tweaked or additional training given before the long-term deployment. This ensures that by the time you implement the protocol fully, it’s already gone through a few feedback cycles. It also prevents discouragement – participants see that their feedback leads to improvements, which can increase buy-in. From a research perspective, an iterative approach helps isolate issues and track improvements, strengthening the validity of conclusions about the protocol’s effectiveness.
* **Maintain and Measure Horizontality:** Ensure that the evaluation itself does not violate the horizontal principles. For instance, if you introduce an **emergency response plan** as part of the protocol (to handle crises with a bit more central coordination), co-develop it with the group and agree on it democratically beforehand. This reflects the concept of democratic centralism: the group agrees on how a temporarily centralized action will be taken in a crisis (template.pdf). During the evaluation, measure how well such mechanisms work. A best practice is to include **governance metrics** or checks—for example, track the proportion of security decisions that are made by group consensus versus unilaterally. If during the long-term study we notice decision-making drifting to a small subset, that’s a red flag that needs addressing (perhaps by adjusting the protocol to involve more people or rotate roles). On the flip side, if the protocol encourages broad discussion on security issues, that’s a success in preserving horizontality. One concrete practice is to use tools like a shared decision log or immutable record (similar to COLBAC’s immutable sphere for logging actions) to enhance transparency (template.pdf) (template.pdf). This way, even if not everyone is directly involved in a particular decision, there is trust that nothing is happening behind closed doors. Incorporating such features and checking their use in the case study is crucial. Essentially, **treat the organization’s participative governance as a “feature” to be protected**: the evaluation should check not only “are threats managed?” but also “is everyone still on board and informed?”.
* **Mitigate Democratic Process Risks:** When bringing security into a democratic context, be aware of the potential for **“democratic attacks”** and plan countermeasures as part of the best practices. For example, set up some form of independent audit or peer review for critical security decisions, which can catch issues like collusion or manipulation early (template.pdf). In the evaluation, one can simulate an attempt at misuse (like someone trying to push through a risky change by exploiting an emergency bypass) and see if the group’s process or the protocol’s rules catch it. The literature suggests measures like dynamic quorum rules (e.g. higher approval percentages required if participation in a vote is low) and limits on emergency actions (template.pdf). Adopting and testing these measures in the case studies is a best practice to ensure the protocol truly holds up under adversarial conditions. It also sends a message to participants that the protocol is robust not only against outside hackers, but also against subtler internal governance attacks – which improves their confidence in the system.
* **Continuous Feedback and Learning:** Finally, treat the evaluation as a two-way learning process. For the organization, it should build security awareness and skills; for the researchers (or framework developers), every observation is a chance to improve the protocol. Regularly share interim findings with the participants, in accessible language, to validate interpretations. This can be done in a wrap-up meeting after each phase, highlighting strengths (e.g. “we identified 3 more threats with the new method than with STRIDE, and participants felt more in control of the process”) and weaknesses (e.g. “the group felt overwhelmed during the voting on access controls”). Such transparency not only adheres to the ethos of the organizations, but also often **yields additional insights** – participants might offer suggestions that researchers didn’t think of. By the end, the evaluation will have not just measured the protocol’s effectiveness, but also refined it and bolstered the organization’s capacity to address threats collectively.

In summary, evaluating a new threat modeling protocol in horizontal organizations requires a blend of rigorous metrics and adaptable, people-centered methods. By using well-defined metrics like usability, precision, recall, efficiency, adaptability, scalability, and horizontality, and by implementing a thoughtful case study methodology, researchers can quantitatively and qualitatively validate the protocol’s effectiveness (template.pdf) (template.pdf). The process must navigate challenges of decentralization with empathy and flexibility, employing best practices that honor the organization’s collaborative nature. The end result is not only an evidence-based assessment of the protocol, but also a set of insights on how security frameworks can successfully integrate with decentralized and democratic teamwork (template.pdf) (template.pdf). Such an approach ultimately contributes to both better security and stronger organizational trust, demonstrating that security and horizontality can indeed go hand in hand.