Towards Agentic AI for Access Control in Cyber-infrastructures: Exploring the Security and Human Factors

[BlueSky Paper]

Carlos Rubio-Medrano
carlos.rubiomedrano@tamucc.edu
Texas A&M University-Corpus Christi
Corpus Christi, Texas, USA

Souradip Nath
snath8@asu.edu
Arizona State University
Tempe, Arizona, USA

Ananta Soneji
asoneji@asu.edu
Arizona State University
Tempe, Arizona, USA

Jennifer Mondragon
jmondragon6@islander.tamucc.edu
Texas A&M University-Corpus Christi
Corpus Christi, Texas, USA

Jaejong Baek
jaejong@asu.edu
Arizona State University
Tempe, Arizona, USA

Gail-Joon Ahn
gahn@asu.edu
Arizona State University
Tempe, Arizona, USA

Abstract

Access Control (AC) remains one of the fundamental paradigms of computer security due to its potential to mitigate serious threats by restricting access to sensitive resources within emerging technologies. However, despite decades of research, the *life-cycle*, i.e., specification, evaluation, and enforcement, of AC policies in modern cyber-infrastructures remains incomplete, inaccurate, and unverified, which largely decreases their effectiveness and efficiency to counteract emerging threats and vulnerabilities.

The recent unparalleled advancements in the field of Generative Artificial Intelligence (AI) present a unique opportunity to address these challenges by introducing AC-Agents, an innovative solution for not only handling the life-cycle of AC policies, but also for collecting relevant threat intelligence and orchestrating the deployment of counteractions in a seamless, automated, and larger scale fashion with a high degree of effectiveness and efficiency.

This paper presents an overview of existing problems affecting modern cyber-infrastructures, discusses potential solutions leveraging emerging AI agents, and elaborates on the security and human factors involved, hoping to inspire a thoughtful discussion within the AC, the security, and the AI communities, such that future approaches can benefit from interdisciplinary collaborations.

CCS Concepts

• **Security and privacy** → **Access control**; *Usability in security and privacy*; • **Computing methodologies** → **Intelligent agents**; *Multi-agent systems.*

Keywords

Agentic Access Control, Policy Specification, Policy Evaluation, Policy Enforcement, Intelligence Collection, Intelligence Orchestration

SACMAT '26, Waterloo, ON, Canada

ACM ISBN 979-8-4007-2107-6/2026/07
https://doi.org/10.1145/3750555.3811881

ACM Reference Format:
Carlos Rubio-Medrano, Souradip Nath, Ananta Soneji, Jennifer Mondragon, Jaejong Baek, and Gail-Joon Ahn. 2026. Towards Agentic AI for Access Control in Cyber-infrastructures: Exploring the Security and Human Factors: [BlueSky Paper]. In *Proceedings of the 31st ACM Symposium on Access Control Models and Technologies (SACMAT '26), July 08–10, 2026, Waterloo, ON, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3750555.3811881

1 Introduction

Access Control (AC) is a fundamental paradigm to enforce cybersecurity properties in modern cyber-infrastructures and emerging technologies, which comprise a plethora of different software, hardware, data, processes, and users. If correctly enforced, AC has the potential to prevent the occurrence of costly security incidents, such as data thefts, privacy leaks, resource misuse, lateral movement, and reconnaissance. These benefits have been recently re-discovered by both industry and academia as a part of Zero Trust [68], a paradigm based on intelligently combining a series of methodologies and techniques largely exercised in the past in isolation, namely, the continuous evaluation and enforcement of authentication, AC and monitoring policies, to enhance the protections currently deployed in practice in the context of modern cyber-infrastructures. The idea is simple: instead of providing classical *perimetral* defenses, e.g., VPNs, firewalls, one-time authentication and authorization, etc., *defense-in-depth* protections are conceptualized and deployed by identifying sensitive resources and service layers, and defining a series of policies that mediate resource access, which can be continuously evaluated over time, often as a result of perceived changes in security-relevant data collected through monitoring. This way, threats like unmediated access to resources, lateral movement, and extended attacker reconnaissance can be better prevented, thus potentially resulting in fewer vulnerabilities and incidents.

Also recently, the field of Generative Artificial Intelligence (Gen-AI) has entered into an unprecedented cycle of interest, to the point it is now regarded as the most disruptive technology of our times, impacting every single field of research and practice, including cybersecurity. When it comes to AC, the use of Gen-AI techniques for developing efficient and effective user-focused tools also becomes an interesting line of opportunity for future work.

With that in mind, in Sec. 2, we start our discussion by introducing the *Unrestricted Movement and Access* Threat Model (UMA-TM), a serious security issue allowing attackers to compromise Research Cyber-infrastructures (RCIs) [63], a set of exemplary cyber-infrastructures composed of a plethora of *heterogeneous* software and hardware components, which are heavily used by *practitioners*, i.e., researchers, students, and IT staff, to develop cutting-edge technological and scientific innovations worldwide. Considering the OSRP framework [58], the UMA-TM targets the integrity and reputation of scientific missions, potentially resulting in loss of intellectual property and the introduction of unverified or altered data, which may have a devastating effect on the public's confidence in scientific endeavors and emerging technologies.

Next, in Sec. 3, we dive into a series of problems preventing the deployment of Zero Trust approaches counteracting the UMA-TM inside RCIs, which include the lack of accurate AC policies, the difficulties in their correct specification, the problems for evaluation and enforcement introduced by the heterogeneous nature of RCIs, as well as the lack of a systematic and consolidated approach to collect, digest, and leverage intelligence and use it towards deploying countermeasures in the form of AC policies.

To address these issues, in Sec. 4, we propose the design, development, and testing of AC-Agents, a solution featuring Gen-AI for the purposes of the specification, evaluation, enforcement, and intelligence collection and orchestration of AC policies within RCIs. For each proposed agent, we elaborate on the expected functionality, the potential benefits, the security research questions and factors to address (Sec. 5), as well as the human factors involved in successfully deploying these innovative technologies to practitioners of RCIs (Sec. 6), which may not be necessarily experts in AC and may certainly benefit from user-centered approaches that handle complexity without sacrificing efficiency and effectiveness.

2 A Threat Model for Cyber-infrastructures

Research Cyber-infrastructures. Modern Cyber-infrastructures are commonly composed of a plethora of software and hardware components, both physical and virtualized, provided by different vendors and implementing various methodologies and techniques, all working towards a common goal. As our running example, consider Research Cyber-infrastructures (RCIs) [63–65], which support scientific research endeavors at a major academic institution [3] through many different *resources* such as computational power, network bandwidth, and data storage. For instance, data-intensive operations are supported by a series of High-Performance Computing (HPC) nodes, e.g., Graphics Processing Units (GPUs). Also, big-data storage is provided by local and remote file systems. For security purposes, *perimetric* defenses such as firewalls and Virtual Private Networks (VPNs), SSH, Browser logins, and Single Sign-On (SSO), are also provided. RCIs are in turn utilized by different types of *Practitioners*, e.g., IT administrators (admins), principal investigators (PIs), post-doctoral researchers (Post-docs), and Students.

Practitioners leverage the many different service components of RCIs to carry out their respective tasks, which consume *sensitive* computational resources as described before. Also, practitioners engage in collaborative projects with each other, namely, researchers

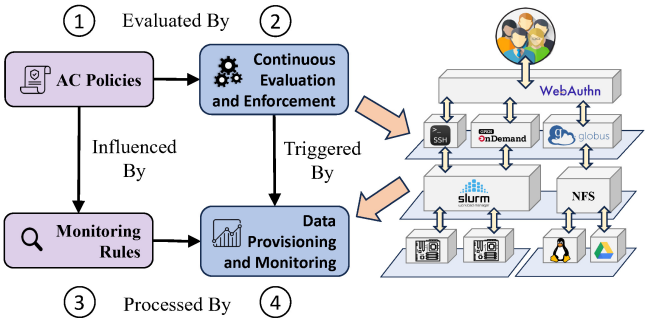

Figure 1: Implementing CAE for RCIs. The UMA-TM is prevented by continuously provisioning, evaluating, and enforcing AC policies (1) (2) triggered as a response to monitoring events consuming data obtained directly from RCIs (3) (4).

from the same or other institutions, who also get access to the aforementioned resources in a trust-based basis. Finally, RCIs may also be the subject of several regulations intended to restrict the functionality and the availability of services for enhanced performance, economy, and security, and are commonly expressed as *Internal*, e.g., team and project-based, *Institutional*, e.g., college-issued, and *Compliance*, e.g., government-issued, AC policies [50].

The UMA-TM. Recent attacks [43, 45] have successfully exploited an *Unrestricted Movement and Access* Threat Model (UMA-TM), in which once *perimetric* defenses, e.g., firewalls and Virtual Private Networks (VPNs), have been bypassed, attackers are given freedom to explore and *move* freely inside RCIs, ultimately obtaining unrestricted access to individual hardware, software components, or sub-systems as a whole [56]. As an example, Credential-Stealing attacks [77] may be used as a preliminary step, allowing for malicious actors to completely bypass existing perimetric defenses. Once inside RCIs, attackers may be allowed to move freely at different levels of service, obtaining full access to resources such as data storage and GPU nodes, which can ultimately result in unintended reconfigurations, installation of bots, etc., as well as the *leak* of sensitive data: personal, research results, business secrets, etc.

3 The Problems with Continuously Enforcing Access Control Policies

As a response to the UMA-TM, proactive countermeasures have been proposed in the form of the Continuous Evaluation and Enforcement of Access Control Policies (CAE) [53]. Fig. 1 provides a graphical depiction of the main components of CAE which prevents attackers from freely *moving* and accessing resources inside RCIs by collecting and communicating security-relevant data from the RCIs themselves, and then using that data to trigger countermeasures to perceived security anomalies in the form of the evaluation and enforcement of AC policies. Whereas the composing elements of CAE has been largely studied in the literature, and have also been implemented in isolation, their use in combination in practice is still in its infancy [9, 28, 68]. Concretely, two recent studies [50, 51], which performed an in-dept exploration of both practitioners and RCIs, identified the problems we discuss next.

3.1 *P-Write*: Writing and Evaluating Policies

RCIs are commonly built on top of a plethora of *heterogeneous* technologies involving a considerable number of resources, processes, and stakeholders. As a result, it becomes effort-costly and error-prone to properly elucidate effective policies, resulting in inadequate or even nonexistent policies at all.

As an example, there is limited support for collaborative policies among researchers from different institutions, resulting in vulnerability-prone policies, or even no policies at all. This *P-Write* problem can be further subdivided into the following sub-problems:

***P-Write*-Model**. There exists no standardized theoretical model or approach that can support the heterogeneous components of RCIs, leaving practitioners with unreliable *ad-hoc* solutions. As an example, whereas Role-based Access Control [74] is informally used in RCIs, it was found to be inconsistently implemented with respect to its reference description [52], which may result in vulnerabilities.

***P-Write*-Stakeholders**. There is also a need to support policies implementing security requirements from multiple *stakeholders*, including sponsors, institutions, IT departments, academic units, research teams, and PIs. However, the policies needed for effective CAE enforcement are only partially known [21]. Also, there is no adequate conflict [25, 36] nor explainability on policy decisions [26].

***P-Write*-Evaluation**. There is a need to support an *always verify* [9] approach based on the continuous evaluation of policies over time, thus effectively enabling the proactive, automated, and usability-aware assignment and revocation of privileges. However, there are no approaches balancing security, effectiveness, and usability when evaluating policies in a continuous manner [10, 21].

3.2 *P-Deploy*: Continuously Enforcing Policies

There are several different users, processes, roles, regulations, stakeholders, as well as a large set of potential institutional policies, unit/departmental policies, team policies, etc., that may be involved in restricting access to resources within RCIs. Moreover, due to the *heterogeneous* nature of RCIs described before, there exists a plethora of different sources for security-relevant data, each of them potentially exhibiting a different level of trust. This makes it difficult to elucidate effective and efficient monitoring rules, such that the mapping between monitoring events and their corresponding *post-actions*, i.e., re-evaluating and re-enforcing policies, can be better assessed and trusted. As an example, in RCIs, privilege revocation is sometimes not enforced correctly when student researchers graduate or are dismissed. The *P-Deploy* problem can be further refined into the following sub-problems:

***P-Deploy*-Enforcement**. There is a need for continuous, automated, and effective enforcement code and settings for decisions taken when evaluating AC policies. However, current reconfigurations to RCIs lack formal verifications [21, 73].

***P-Deploy*-Provisioning**. There is a need for the provisioning of policy- relevant data from the RCIs themselves and related sources. However, there exists no convenient and automated approaches addressing such a requirement, as current implementations rely on manual, self-assessed solutions, e.g., XACML [5].

***P-Deploy*-Monitoring**. There is also a need for monitoring security-relevant data based on well-defined rules that include, among other things, the time of provisioning, and *pre* and *post* rule conditions, for example, triggering policy evaluation when a given rule is activated. However, existing solutions provide no formal approaches supporting monitoring rules in RCIs [8].

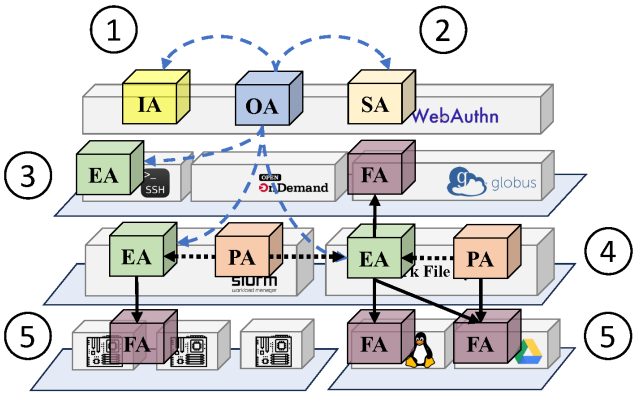

Figure 2: AC-Agents for RCIs. Intelligence on emerging threats (IAs) (1) is used to coordinate and deploy countermeasures (OAs) in the form of the specification of new policies (SAs) (2) and/or the evaluation (EAs) (3), the provisioning (PAs) (4), and enforcement (FAs) (5) of existing ones. AC-Agents may be placed in different layers of service within RCIs.

3.3 *P-Intel*: Intelligence and Counteraction.

Finally, it is also difficult to identify, collect, process, and properly react to threat intelligence information for RCIs, in an effort to better protect the security of the overall infrastructure by deploying authorization policies as needed. As an example, addressing newly-discovered vulnerabilities may take weeks or months before a countermeasure is tested and deployed within a given RCI.

***P-Intel*-Collect**. There is a need to locate effective and trustful intelligence information relevant to RCIs, which, besides requiring extended expertise on the matter, may be also time-consuming and error-prone due to the many different sources involved, the data formats, and the levels of perceived trust [82].

***P-Intel*-Process**. Also, there is a need to dissect, select, and evaluate intelligence information for RCIs in a prompt and efficient manner, a process that may also require extended expertise and considerable time and effort, eventually becoming error-prone [82].

***P-Intel*-Counteract**. Finally, there is also a need to identify, design, and orchestrate candidates for security countermeasures in the form of authorization policy updates and reconfigurations in an efficient and prompt manner, thus potentially reducing the risk of successful vulnerability exploits and incidents [54].

4 AC-Agents: Leveraging AI for Access Control

Our approach towards addressing the problems just described is based on *cognitive partnership* [14], a philosophical stance in which Gen-AI agents assist and enhance human capabilities, e.g., by summarizing knowledge, suggesting solutions, automating manual work, etc. This idea follows emerging frameworks for Gen-AI agents such as A^2C (automated, augmented, collaborative) [7], or the 5-level autonomy: Decision Support, Human Approval, Human

Veto, Execute and Inform, and Fully Automated [67], in which humans, e.g., the *practitioners* of RCIs, remain in full control, as they evaluate suggestions, request changes, and provide a final approval.

With that in mind, we envision AC-Agents, an approach leveraging emerging agentic AI techniques to support the deployment of CAE solutions in modern cyber-infrastructures, e.g., RCIs. AC-Agents graphically shown in Fig. 2, will assist practitioners on *specifying* (SAs), *provisioning* (PAs), *evaluating* (EAs), *enforcing* (FAs), effective, efficient, and usable AC policies. In addition, AC-Agents will collect *intelligence* (IAs) on security vulnerabilities and will use it to *orchestrate* (OAs) other AC-Agents to deploy CAE cyber-protections for RCIs. This way, our proposed approach will assist on counteracting the UMA-TM and other emerging threats, thus promoting science developments and discoveries by enforcing FAIR (Findable, Accessible, Interoperable, and Reusable) principles [90].

4.1 Agents for Specifying Policies (SAs)

Specification AC-Agents (SAs) will assist on generating and translating AC policies from different sources and regulations, striving to provide a convenient and seamless approach for practitioners that ultimately results in an enhanced level of confidence and trust, which will be crucial to overcome *P-Write*-Model and *P-Write*-Stakeholders. For instance, researchers may use SAs to write policies restricting access to sensitive personal data collected as a part of their projects while meeting institutional and compliance requirements on the matter. Overall, SAs may support the following:

Natural Language Translation. SAs will assist on translating Natural Language (NL) policies, e.g., institutional or regulatory policies describing the handling of sensitive data, into well-defined formats, e.g., XACML, which could be then processed elsewhere, e.g., by dedicated XACML APIs providing evaluation engines. Recent works have explored this area of research [4, 13, 60, 81, 89, 92].

Interactive Generation. SAs will provide guided sessions allowing for practitioners to elucidate AC rules and policies in an informative and interactive way. These sessions may in turn leverage pre-define policy templates or suggestions to make the process more efficient [11, 47], and may also leverage the policy translation features described next to improve understanding and confidence on the results. Recent work has started exploring this idea [48].

Resolution of Specification Conflicts. Finally, SAs will provide guidance on potential decision conflicts and their consequences that may arise when policies are being developed [19, 93]. For instance, SAs should alert users when a newly-crafted policy may render a given resource completely inaccessible to a set of practitioners working together in a collaborative project. Once a conflict is identified, a set of potential solutions must be presented to practitioners This feature may be combined with the interactive generation and the policy explanation features just described.

Explainability. SAs will provide concise natural language explanations and summarizations of AC policies, allowing for practitioners to understand policies while policy crafting is taking place, as just explained or as a part of a subsequent analysis [60].

4.2 Agents for Evaluating Policies (EAs)

Evaluation AC-Agents (EAs) will store, retrieve, and evaluate AC policies provided by SAs, and will also combine, explain, and resolve the conflicts originated by policy evaluation decisions, thus addressing *P-Write*-Evaluation by supporting the following features:

Policy Storage and Retrieval. EAs will process and store internally policies, and will also identify and retrieve policies that are relevant to a given access request, incorporating Gen-AI techniques such as Retrieval Augmented Generation (RAG) [15] and Reinforcement Learning (RF) [61] to allow for policies to be stored internally and retrieved with a high degree of efficiency [37].

Policy Data Provisioning. EAs will automatically identify the data enlisted in the AC policies that are relevant to a given access request, and will *provision* it, e.g., identify, format, and communicate it, directly from RCIs. Such a feature may in turn benefit from contacting our proposed PAs introduced in Sec. 4.4, allowing for inter-communicating agents to organize into *marketplaces* [62] for secure and trusted data provisioning within RCIs.

Policy Evaluation. EAs will also evaluate and combine the decision results of AC policies relevant to a given access request implementing different strategies for evaluation order and precedence, leveraging both the internal structure of policies as well as any relevant contextual information. For instance, EAs may give preference to institutional and compliance regulatory policies when evaluating access requests, as they may have a greater impact on a final combined access decision, making their evaluation a priority over personal policies issued by practitioners [41, 48, 59, 75, 79].

Resolution of Evaluation Conflicts. EAs will identify, prepare, and suggest resolution strategies for conflicting policy decisions based both on strategies identified previously in the literature [31] as well a those relevant in the context of RCIs. As an example, conflict resolution strategies may prioritize deny decisions originated from institutional and compliance policies over the rest.

Explainability. Complementing the features just discussed, EAs will also provide concise and didactic explanations on the decisions obtained when evaluating AC policies, allowing for practitioners to better understand why a given access request was either authorized or denied in the presence of several policies and data [26, 39, 72, 98].

4.3 Agents for Enforcing Policies (FAs)

Enforcement AC-Agents (FAs) will continuously enforce AC policies as dictated by EAs via automated RCI re-configurations, e.g., script and code generation, thus addressing *P-Deploy*-Enforcement.

Automated Re-configurations. FAs will automatically generate enforcement code and/or configuration scripts for the many different technologies involved in RCIs, thus alleviating practitioners from having to become highly skilled in the inner workings and details of such technologies. Also, the potential for errors and the unwanted introduction of security vulnerabilities may be better prevented, as there may no longer be a need to write code or scripts from scratch. Recent work has started exploring this idea [69].

Explainability. Finally, FAs will also provide explainability support at the enforcement level, providing practitioners with concise and didactic NL accounts on what policy decisions were enforced, what changes were made, what technologies were affected, etc. Interactive sessions will be supported, allowing practitioners to obtain detailed explanations, encouraging confidence and trust.

4.4 Agents for Provisioning Data (PAs)

Provisioning AC-Agents (PAs) will collect and validate security-relevant data directly from RCIs, will implement and evaluate a series of data monitoring rules, and will communicate the collected data to EAs and FAs as a result of monitoring events or upon request, thus addressing *P-Deploy*-Provisioning and *P-Deploy*-Monitoring.

Data Provisioning. PAs will implement repositories to identify and store information pertaining security-relevant data within RCIs, such that it can be later collected at runtime as a response to policy evaluations directed by EAs, policy enforcement procedures directed by FAs, or as a result of executing data monitoring rules. PAs will also provide the means to locate, format, and communicate security-relevant data dynamically from RCIs following runtime strategies based on convenience and performance efficiency [71].

Rule Processing. PAs will leverage the repositories and the provisioning strategies just discussed to support the specification and evaluation of data monitoring rules, which, besides consuming data obtained from RCIs, will also implement a series of strategies to respond to monitoring events. Such strategies may include triggering the evaluation and/or enforcement of AC policies, the specification of new policies by invoking SAs, or even the collection and deployment of intelligence counteracting measures as provided by IAs and OAs as it will be described later in this paper. As an example, a monitoring rule may be set to detect when student researchers graduate from a given institution. When activated, the rule may trigger the invocation of EAs and FAs to remove any access rights assigned to the student in the RCI. To prevent potential security vulnerabilities and misuse, PAs may implement an administrative AC approach to regulate what rules can be specified and enforced [18].

Explainability. Finally, as with SAs, PAs will provide concise explanations detailing the procedures of data provisioning, as well as the nature of the evaluation of monitoring rules at specification time, allowing for practitioners to identify potential consequences.

4.5 Agents for Collecting Intelligence (IAs)

Intelligence AC-Agents (IAs) will identify, collect, assess, and process intelligence information on existing or potential threats to RCIs from different sources, thus addressing *P-Intel*-Process.

RCI Modeling. IAs will maintain a model representation depicting the inner structure of RCIs, e.g., software, hardware, layer of service, processes supported, practitioners, etc. This way, IAs may be able to identify the intelligence information relevant for a given RCI based on the technologies and usage patterns involved.

Assessment and Processing. IAs will then combine different pieces of intelligence information, and will assess them against the internal RCI model, a series of security and safety properties, and practitioner input, allowing for an interactive V&V process. As an example, information pertaining to a newly-discovered zero-day vulnerability may be provided by several trusted sources with different levels of detail. IAs will identify the information, assess the trust level of each source, combine the different pieces altogether, and determine its relevance against the RCI they serve. Later, practitioner input, i.e., IT staff, will be procured for further countermeasure action by passing over the information to OAs.

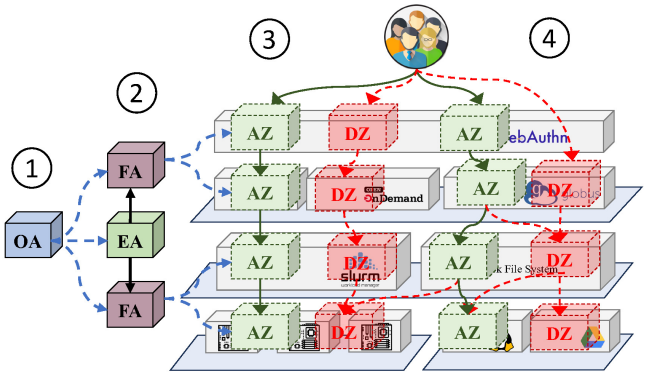

Figure 3: Orchestrating access paths within RCIs. OAs (1) interact with EAs and FAs to automatically create either *authorized* or *denied* zones on-the-fly (2), thus providing practitioners with temporal, case-by-case access to RCIs resources (3) (4).

Explainability. In order for practitioners to understand and validate intelligence information, IAs must provide concise, interactive, and didactic explanations, accommodating different levels of detail and providing illustrative examples when appropriate. This way, different levels of expertise, as depicted by practitioners, e.g., expert IT staff vs novice student researchers, may be better served.

4.6 Agents for Orchestrating Protections (OAs)

Orchestration AC-Agents (OAs) will process the information provided by IAs to recommend potential countermeasures to practitioners, which may be then deployed inside RCIs via the *orchestration*, i.e., the synchronized communication of data, commands, and responses with other AC-Agents, thus addressing *P-Intel*-Counteract.

Countermeasure Planning. OAs will elucidate a series of *orchestration strategies* to effectively deploy countermeasures based on AC policies as a response to both intelligence information and feedback provided by practitioners. As an example, Fig. 3 shows a countermeasure strategy in which OAs coordinate with other AC-Agents to dynamically create *access paths* within RCIs composed of *authorized* (AZ, green) and *denied* (DZ, red) access zones, which are in turn created on-the-fly by evaluating and enforcing AC policies. This way, practitioners will only *traverse green* access paths while all other *red* will remain inaccessible, thus preventing the lateral movement depicted by the UMA-TM discussed before.

Agent Orchestration. OAs will communicate with other AC-Agents to deploy countermeasure strategies approved by practitioners inside RCIs. That will include the synchronized communication of data, commands, and responses on the evaluation and enforcement of AC policies following the high-level orchestration strategies just discussed. In addition, a series of *safe cleanup* strategies will also be implemented to account for unexpected errors at runtime. For instance, an orchestration strategy may account for different FAs creating green AZs as shown in Fig. 3. However, in case one of those FAs fails to effectively create a given AZ due to a runtime error, thus preventing the successful creation of an access path, a cleanup strategy may need to *close* all other AZs in the access path, e.g., by turning back each green AZ in the path into a red DZ.

Explainability. Finally, as with other AC-Agents, the success of the features devised for OAs will depend heavily in a series of explainability capabilities that can effectively engage practitioners and capture their intentions in a clear, unambiguous way, allowing them to fully understand the nature of the countermeasures, the cyber-protections provided, and any potential consequences.

5 Security Factors in Agentic AI for RCIs

Having described our proposed AC-Agents, we now elucidate the factors that must be taken into account for providing security guarantees when deploying to RCIs. For each factor, we provide a description, an illustrative example, and guidelines for future work.

5.1 Trust Assessment and Management

Policy Sources. As described in Sec. 4.1, our SAs will leverage a series of AC policies for the purposes of serving as training data for their backend knowledge repositories, e.g., LLMs. Also, such policies may serve as templates for potential recommendations to practitioners. Given their importance, these policies must be therefore analyzed, formally or informally verified, and vetted by experts in the field before they can be consumed by SAs and later by other AC-Agents, following a well-defined model that allows for trust to be quantified, assessed, and later used for making vetting decisions. As an example, as described in Sec. 2, several different NL institutional and compliance policies regarding RCIs must be supported, alongside a large variety of collaborative, team-based, and individual ones. Such policies must therefore be analyzed for correctness, suitability, coverage, etc., allowing for trust to be assessed uniformly during the whole life-cycle of policies [6, 66].

Data Provisioning Sources. Following Sec. 4.4, different software and hardware components within RCIs may eventually serve as sources for security-relevant data to be consumed later by EAs and FAs. As with input policies, the degree of trust and confidence depicted by these sources must be continuously assessed following a well-defined model before they can be incorporated as a part of AC-Agents, preventing attackers from manipulating AC-Agents and RCIs as a consequence of compromising their integrity. As an example, runtime data from a software component may be consumed by EAs. However, if later a zero-day vulnerability allowing for unwanted remote manipulation is discovered, trust-driven countermeasures may preserve the integrity of policy evaluations. Recent work has explored this idea [70, 76].

Intelligence and Countermeasure Sources. Following Sec. 4.5, there is a need to assess a degree of confidence and trust on the different sources, e.g., websites or other agents, the intelligence information consumed by IAs will be extracted from, allowing to incorporate practitioner feedback to decide if a given piece of information as provided by a source can be safely and securely incorporated into a subsequent analysis process. As an example, following our previous description on trusting data sources, a trust model must be in place for continuously assessing the source of the intelligence information on new zero-day vulnerabilities, such that further actions can be taken as a result by AC-Agents. Recent work has started exploring this idea [35, 83].

5.2 Verifying Security Properties

Generated Policies. Following Sec. 4.1, there is a need to verify that the policies generated by SAs preserve institutional, departmental, team, and individual security properties. As an example, practitioners may want to enforce a security property stating they are the only ones having access to sensitive data within the context of a given project. When suggesting and confirming new recommended policies regarding access to such data, SAs must verify that the aforementioned property is preserved. Overall, elucidating these properties in the context of multiple practitioners engaging in collaborative resource-sharing within RCIs represents a non-trivial problem. Recent work has explored related solutions [2, 29].

Evaluation Decisions. There is also a need to verify that the policy evaluation decisions, as produced by EAs in the presence of single or multiple rules and policies, are correct with respect to a reference description or implementation. Potential pre-generation approaches in such regard may include curating training policy data when building backend LLMs, or performing an RL-inspired process based on a series of case scenarios. In addition, post-generation approaches may include leveraging theorem provers [32, 40].

Enforcement Code. Conversely, there is a need to verify the correctness of the generated code and scripts for policy evaluation as prepared by FAs against a series of security and behavioral properties, ensuring that policy decisions are correctly enforced and no side-effects are introduced, thus preventing potential vulnerabilities and inconveniences. As an example, researchers may want to restrict access to project data for members of their collaborative team only, and receive confirmation obtained directly from RCI configurations that no one else can access such data after a series of policy-driven updates. Recent work has leveraged the use of formal specifications for verifying that LLMs correctly generate security-sensitive and security-related code [69].

Orchestration Plans. Finally, there is also a need to verify the plans generated by OAs against a series of security and safety properties, ensuring that effective cyber-protections are deployed and no vulnerability-prone side effects are created. As an example, the access paths graphically shown in Fig. 3 must be contained to resources related to the original access request as stated by practitioners, thus preventing the UMA-TM shown in Sec. 2.

5.3 Securing and Safeguarding AC-Agents

Access Control for AC-Agents. As AC-Agents may process untrusted inputs, they may be expos to attacks such as indirect prompt injection [23], which could be further exploited to escalate privileges [96]. Therefore, it is critical to regulate the privileges granted to these agents through administrative AC models, in line with established principles such as least privilege, mandatory access control, and separation of duties [94]. Generating correct and secure pre-defined policies for AC-Agents is challenging, as it requires significant expertise, and static policies may fail to adapt to the dynamic nature of tasks [97]. To address this, specialized, isolated expert models that operate only on trusted inputs may be used to generate on-demand, least-privilege policies. Recent work has begun exploring such frameworks [79, 86]. Furthermore, these

expert models may continuously improve by generalizing from policies curated by experienced administrators [91].

Secure Agent-to-Agent Communication. As described in Sec. 4, the application of AI agents to support CAE in modern cyber-infrastructures is envisioned as a multi-agent system (MAS) [20], in which specialized, task-specific agents communicate and coordinate with one another to achieve their individual objectives while collectively pursuing the shared goal of protecting the infrastructure from threats. For example, while evaluating policies, EAs may identify gaps in the policies or inconsistencies in provisioning data and communicate with the SAs and FAs, respectively, to dynamically address these issues. Therefore, it is critical to safeguard these agent-to-agent communications through secure orchestration mechanisms to prevent AI agents themselves from becoming potential attack vectors [42]. With agent-to-agent communication becoming increasingly (Google's A2A protocol[1]), securing these interactions and mitigating emerging risks becomes crucial [33, 38, 84].

Social Engineering Attacks. Gen-AI systems may exhibit sycophancy or over-alignment, disproportionately favoring user agreement over accuracy and critical evaluation [78]. Hence, our AC-Agents must undergo bias and fairness evaluation across user roles and operational contexts to ensure equitable and consistent decision support [1]. AC-Agents may be also susceptible to external social engineering attacks, including prompt injection, adversarial manipulation, and misuse through deceptive inputs [1, 27]. Accordingly, AC-Agents must occur within secured environments with safeguards against unauthorized access, model extraction, prompt injection, and related adversarial threats [80]. It is also advisable to have clear documentation to describe system capabilities, limitations, and appropriate use boundaries by stakeholders [22].

Ethics and Responsible Gen-AI. Given the security-sensitive nature of tasks to be supported by AC-Agents, their ethical and responsible use is paramount [34, 44]. First, AC-Agents must not be operated without oversight and all inputs/outputs will require human review and approval before implementation [57]. Next, to ensure reliability, outputs must be validated against expert opinions and systematically tested against adversarial inputs to prevent manipulation, prompt injection, and model exploitation risks [23]. Comprehensive audit logs must be maintained to document prompts, outputs, model versions, and human decisions, enabling accountability, traceability, and post-hoc review. Sensitive raw data of cyber-critical infrastructures should not be directly shared with AC-Agents and other Gen-AI tools, and data shared with such tools must follow strict data minimization principles [17]. Where appropriate, de-identified or simulated data could be shared through secure APIs or locally deployed models within controlled environments. Finally, data retention and deletion policies must be enforced to prevent unnecessary persistence. All models and configurations must be version-controlled, with documented update procedures and rollback mechanisms to preserve reproducibility and stability [88].

6 Human Factors in Agentic AI for RCIs

Having explored the potential security factors to consider in AC-Agents, we now provide a non-comprehensive exploration of the human-centered factors that may be relevant when developing AC-Agents for RCIs. For each factor, we provide a brief description, an illustrative example, and some guidelines for future work.

6.1 Prioritizing Convenience

Training. Successful implementations must take into account the *cognitive load*, i.e., the knowledge needed as well as the mental effort involved in using AC-Agents effectively when interacting with RCIs. In such regard, convenient training approaches, which can be easily understood and assimilated by practitioners, will be required. Also, successful implementations must provide a convenient mental operational model, which allows for practitioners to easily grasp the inner workings of AC-Agents without having to receive additional assistance and/or obtain extra information frequently, preferably resorting to background foundational concepts that may be commonly understood by practitioners as a part of their daily-life routines. For instance, practitioners engaging in collaborative research teams should be allowed to specify their own resource-sharing AC policies without the need to become extensively knowledgeable in AC theoretical models, e.g., RBAC, ABAC, etc., either in isolation or in combination. Instead, they should leverage convenient and easy-to-grasp operational models that can better reflect their AC needs and can be easily understood and practiced without extensive training and support [49, 51].

Explainability. As featured before in this paper, AC-Agents must provide explanations informing of the different operations performed or to be performed, allowing for practitioners to make decisions and stay in control of the overall cyber-protection enforcement process, following the cognitive partnership approach discussed in Sec. 4. In such regard, future approaches providing effective explanations must take into account the ongoing operational context, the set of AC policies involved, the technologies being accessed/shared, as well as the potential results and consequences of the operations being considered. As an example, practitioners providing access to resources to a new collaborator must be informed of the resources affected, the relevant AC policies, the access rights to be updated, as well as any consequences involved, including potential resource misuse. The AC community has recently identified the importance of addressing this topic [26, 72, 98].

Interactivity. Complementing explanations, AC-Agents must also provide interactive sessions that can provide a clear operational sequence that guides practitioners through a series of well-defined steps, allowing for pausing, resuming, backtracking, restarting, or even canceling the whole process if needed. As with the *Training* factor described before, the use of convenient mental models may be crucial for providing effective and responsive interactive sessions, reducing the mental workload of practitioners, and resulting in improved usability and satisfaction. In such regard, future approaches may benefit from a combination of text, voice-based, and graphical interfaces, such that information can be visualized and digested in a convenient and effective way by practitioners. As an example, future approaches may combine textual/voice NL explanations with some graphical representations of RCIs, e.g., the ones featured in Fig. 3, allowing for practitioners to visualize the resources affected by AC-related operations, and suggest changes by means of NL commands or by interacting with a graphical interface [24, 51].

[1] https://a2a-protocol.org/latest/

Personalization. AC-Agents may also benefit from providing personalization features for practitioners, e.g., capturing their preferences, any relevant contextual information such as administrative roles, e.g., IT staff, PI, or student, as well as a recount of previous interactions. Personalization features may also be helpful to adjust the level of technical details provided as a part of explanations and interactive features with the expertise possessed by practitioners, allowing for a more convenient experience. As an example, personalization features may allow practitioners to quickly add or remove a collaborator from a resource-sharing scheme by analyzing previous interactions to suggest the resources as well as the access rights involved in the operation, thus preventing missteps [55, 87, 95].

6.2 Guaranteeing Safety

Privacy. Privacy constitutes a very important topic towards guaranteeing the secure, safe, and fair use of existing and emerging technologies [16]. When it comes to our proposed AC-Agents, maintaining privacy guarantees for practitioners may be crucial to build trust and confidence not only as a result of deploying cyber-protections as discussed before in this paper, but also as a result of adequately handling all personal and/or sensitive data obtained from practitioners [12, 30, 46]. In such regard, privacy considerations may include not internally storing nor communicating personal data with other agents (including AC-Agents) without proper disclosure and explicit authorization from practitioners. As an example, the personalization and explainability features described before must refrain from consuming and disseminating personal and sensitive data as a part of their expected recommendations.

Transparency. AC-Agents must provide transparency guarantees by disclosing all the information related to features they will provide, including base knowledge, the reasoning steps involved in their *thinking process*, as well as a recount of any operations performed on RCIs. In addition, they must also clearly state their theoretical and/or operational limitations, and inform practitioners when certain tasks cannot be completed in an efficient, safe, and secure way, suggesting alternative actions instead. For instance, following Fig. 3, `OAs` may try to create alternative access paths in case the RCI configurations make the initial proposal unfeasible.

Scrutability. Following the need for transparency, AC-Agents must also allow practitioners to clearly state their satisfaction or dissatisfaction with respect to the features and explanations presented, especially when interacting with them via NL prompts [26]. This must not only allow for AC-Agents to design, suggest, and implement potential alternative courses of action, but should also have no unpleasant side effects or any potential changes that compromise both security and efficiency. For instance, when suggesting potential AC policies, `SAs` must refrain from suggesting *permissive* policies granting more resource access rights than needed in an attempt to please a practitioner who has raised negative feedback.

Escalation. Complementing the discussion on transparency and scrutability, AC-Agents must also escalate to a human operator, i.e., an IT staff member, when a given operation proves to be unfeasible due to internal limitations, either theoretical or practical, or also because of the lack of positive responses from the practitioner they serve, possibly caused by a dissatisfaction with the suggested alternatives presented by the agent. For example, recent work has incorporated escalation within LLM-based policy generation [37].

6.3 Improving Performance

Side Effects. AC-Agents must refrain from introducing undesired side effects that have a negative impact in the workload experienced by practitioners, either in the work performed by the AC-Agents themselves or in the research duties that are facilitated by RCIs. As an example, `FAs` must not introduce policy enforcement re-configurations that notoriously slow down the runtime performance of scientific calculations that leverage certain hardware and software products within RCIs. In another example, `SAs` must strive to specify *efficient* AC policies that are not time and resource-consuming when it comes to being evaluated at runtime by `EAs`, thus preventing potential performance affectations to time-sensitive scientific calculations. The nature and the way to create such policies remain as an interesting challenge for future work.

Effort Reduction. There is a need for approaches and metrics to quantify the human effort required by practitioners to carry out security and AC-related duties inside RCIs, either with and without the support of AC-Agents, such that potential improvements resulting from a significant effort reduction can be assessed and used to justify future improvements as well as eventual adoptions in practice. As an example, metrics are needed to quantify the time and effort involved in the specification of AC policies [50].

Trust and Confidence. Finally, there is also a need to develop metrics to precisely quantify the levels of trust and confidence as experienced by practitioners with respect to AC-Agents. Such data may be crucial not only for the development of enhanced Gen-AI tools for security purposes, but also to encourage the successful deployment of these solutions in practice, providing communities of practice featuring AC-Agents, to complement successful case studies with data obtained directly from production environments.

7 Conclusions and Future Work

In this paper, we have proposed AC-Agents, an approach leveraging emerging Gen-AI techniques to protect RCIs via AC policies. We have also discussed a series of potential security and human factors involved in developing AC-Agents, which must be addressed in the upcoming years. As a part of our ongoing work, we are exploring the development of `SAs` [49], `EAs` [48], and `FAs` [69], and plan to start developing `PAs`, `IAs`, and `OAs` in the near future. We expect our AC-Agents to become available via the Apache 2.0 open-source license [85], thus encouraging institutions implementing RCIs to develop, test, and share them as a part of a community of practice. Finally, we believe our AC-Agents can be potentially deployed in other emerging technologies, e.g., Connected Automated Vehicles and Unmanned Aerial Systems, which may also allow end-users to handle cyber-protections in a convenient way.

Acknowledgments

This work was partly supported by the National Science Foundation (NSF) under grant NSF-CICI-2232911 and by the Institute of Information & Communications Technology Planning & Evaluation (IITP) through grants: RS-2024-004398199 and RS-2024-00442085.

References

[1] Sara Abdali, Richard Anarfi, Carlos J Barberan, Jia He, and Erfan Shayegani. 2024. Securing large language models: Threats, vulnerabilities and responsible practices. *arXiv preprint arXiv preprint arXiv:2403.12503* (2024).

[2] Edoardo Allegrini, Ananth Shreekumar, and Z Berkay Celik. 2025. Formalizing the Safety, Security, and Functional Properties of Agentic AI Systems. *arXiv preprint arXiv:2510.14133* (2025).

[3] Arizona State University. 2024. Large-scale computational resources. https://cores.research.asu.edu/research-computing/about.

[4] Vijayalakshmi Atluri. 2025. Policy Mining: Putting LLMs to Work [Keynote Abstract]. In *Proc. of the 30th ACM Symp. on Access Control Models and Technologies (SACMAT '25)*. Association for Computing Machinery, New York, NY, USA, 1–2.

[5] AT&T. 2021. XACML-3.0. https://github.com/att/xacml-3.0. [Online; accessed June-20-2021].

[6] Juliana Barbosa, Eduarda Alencar, Grace Fan, Aécio Santos, and Juliana Freire. 2025. HILTS: Human-LLM collaboration for effective data labeling. *Information Systems* (2025), 102660.

[7] Mohan Baruwal Chhetri, Shahroz Tariq, Ronal Singh, Fatemeh Jalalvand, Cecile Paris, and Surya Nepal. 2024. Towards human-ai teaming to mitigate alert fatigue in security operations centres. *ACM Transactions on Internet Technology* 24, 3 (2024), 1–22.

[8] Jim Basney, Heather Flanagan, Terry Fleury, Jeff Gaynor, Scott Koranda, and Benn Oshrin. 2019. CILogon: Enabling Federated Identity and Access Management for Scientific Collaborations. *Proc. of Science* 351 (2019), 031.

[9] Elisa Bertino. 2021. Zero trust architecture: does it help? *IEEE Security & Privacy* 19, 05 (2021), 95–96.

[10] Christoph Buck, Christian Olenberger, André Schweizer, Fabiane Völter, and Torsten Eymann. 2021. Never trust, always verify: A multivocal literature review on current knowledge and research gaps of zero-trust. *Computers & Security* 110 (2021), 102436.

[11] Kaiyan Chang, Songcheng Xu, Chenglong Wang, Yingfeng Luo, Xiaoqian Liu, Tong Xiao, and Jingbo Zhu. 2024. Efficient prompting methods for large language models: A survey. *arXiv preprint arXiv:2404.01077* (2024).

[12] Chaoran Chen, Daodao Zhou, Yanfang Ye, Toby Jia-jun Li, and Yaxing Yao. 2025. Clear: Towards contextual llm-empowered privacy policy analysis and risk generation for large language model applications. In *Proc. of the 30th Int. Conf. on Intelligent User Interfaces*. 277–297.

[13] Ye Cheng, Minghui Xu, Yue Zhang, Kun Li, Hao Wu, Yechao Zhang, Shaoyong Guo, Wangjie Qiu, Dongxiao Yu, and Xiuzhen Cheng. 2025. Say what you mean: Natural language access control with large language models for internet of things. *arXiv preprint arXiv:2505.23835* (2025).

[14] Katherine M Collins, Ilia Sucholutsky, Umang Bhatt, Kartik Chandra, Lionel Wong, Mina Lee, Cedegao E Zhang, Tan Zhi-Xuan, Mark Ho, Vikash Mansinghka, et al. 2024. Building machines that learn and think with people. *Nature human behaviour* 8, 10 (2024), 1851–1863.

[15] Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The power of noise: Redefining retrieval for rag systems. In *Proc. the 47th International ACM SIGIR Conf. on Research and Development in Information Retrieval*. 719–729.

[16] Badhan Chandra Das, M Hadi Amini, and Yanzhao Wu. 2025. Security and privacy challenges of large language models: A survey. *Comput. Surveys* 57, 6 (2025), 1–39.

[17] Saswat Das, Jameson Sandler, and Ferdinando Fioretto. 2025. Beyond Jailbreaking: Auditing Contextual Privacy in LLM Agents. *preprint arXiv:2506.10171* (2025).

[18] MAC Dekker, Jason Crampton, and Sandro Etalle. 2008. RBAC administration in distributed systems. In *Proc. of the 13th ACM SACMAT*. 93–102.

[19] Beibei Fan, Xiaoyan Liang, Yang Luo, Yang Bo, and Chunhe Xia. 2011. Conflict detection model of access control policy in collaborative environment. In *2011 Int. Conf. on Computational and Information Sciences*. IEEE, 377–381.

[20] Jacques Ferber and Gerhard Weiss. 1999. *Multi-agent systems: an introduction to distributed artificial intelligence*. Vol. 1. Addison-wesley Reading.

[21] Eduardo B Fernandez and Andrei Brazhuk. 2024. A critical analysis of Zero Trust Architecture (ZTA). *Computer Standards & Interfaces* 89 (2024), 103832.

[22] Yuyou Gan, Yong Yang, Zhe Ma, Ping He, Rui Zeng, Yiming Wang, Qingming Li, Chunyi Zhou, Songze Li, Ting Wang, et al. 2024. Navigating the risks: A survey of security, privacy, and ethics threats in llm-based agents. *arXiv preprint arXiv:2411.09523* (2024).

[23] Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. 2023. Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection. In *Proc. of the 16th ACM workshop on artificial intelligence and security*. 79–90.

[24] Gelareh Hasel Mehri, Tien Dung Le, Bram Cappers, Jerry Den Hartog, and Nicola Zannone. 2024. WiP: Enhancing the Comprehension of XACML Policies. In *Proc. of the 29th ACM SACMAT*. 41–46.

[25] Gelareh Hasel Mehri, Benjamin Monmege, Clara Bertolissi, and Nicola Zannone. 2024. A bargaining-game framework for multi-party access control. In *Proc. of the 29th ACM SACMAT*. 127–138.

[26] Gelareh Hasel Mehri, Charles Morisset, and Nicola Zannone. 2025. Towards Explainable Access Control (BlueSky Paper). In *Proceedings of the 30th ACM Symposium on Access Control Models and Technologies*. 117–126.

[27] Feng He, Tianqing Zhu, Dayong Ye, Bo Liu, Wanlei Zhou, and Philip S Yu. 2025. The emerged security and privacy of llm agent: A survey with case studies. *Comput. Surveys* 58, 6 (2025), 1–36.

[28] Yuanhang He, Daochao Huang, Lei Chen, Yi Ni, and Xiangjie Ma. 2022. A survey on zero trust architecture: Challenges and future trends. *Wireless Communications and Mobile Computing* 2022, 1 (2022), 6476274.

[29] Yifeng He, Ethan Wang, Yuyang Rong, Zifei Cheng, and Hao Chen. 2025. Security of ai agents. In *2025 IEEE/ACM Int. Workshop on Responsible AI Engineering (RAIE)*. IEEE, 45–52.

[30] Xinyi Hou, Yanjie Zhao, and Haoyu Wang. 2025. On the (in) security of llm app stores. In *2025 IEEE Symposium on Security and Privacy (SP)*. IEEE, 317–335.

[31] Hongxin Hu, Gail-Joon Ahn, and Ketan Kulkarni. 2011. Anomaly discovery and resolution in web access control policies. In *Proc. of the 16th ACM Symp. on Access control models and technologies*. ACM, 165–174.

[32] Vincent C Hu, Rick Kuhn, Dylan Yaga, et al. 2017. Verification and test methods for access control policies/models. *NIST Special Pub.* 800, 192 (2017), 800–192.

[33] Ken Huang, Yasir Mehmood, Hammad Atta, Jerry Huang, Muhammad Zeeshan Baig, and Sree Bhargavi Balija. 2025. Fortifying the Agentic Web: A Unified Zero-Trust Architecture Against Logic-layer Threats. *arXiv preprint arXiv2508.12259* (2025).

[34] Ken Huang, Vineeth Sai Narajala, John Yeoh, Jason Ross, Ramesh Raskar, Youssef Harkati, Jerry Huang, Idan Habler, and Chris Hughes. 2025. A novel zero-trust identity framework for agentic AI: Decentralized authentication and fine-grained access control. *arXiv preprint arXiv:2505.19301* (2025).

[35] Yukun Huang, Sanxing Chen, Hongyi Cai, and Bhuwan Dhingra. 2024. To Trust or Not to Trust? Enhancing Large Language Models' Situated Faithfulness to External Contexts. *arXiv preprint arXiv:2410.14675* (2024).

[36] Trent Jaeger, Reiner Sailer, and Xiaolan Zhang. 2004. Resolving constraint conflicts. In *Proc. of 9th ACM symp. on Access control models and technologies*. 105–114.

[37] Sakuna Harinda Jayasundara, Nalin Asanka Gamagedara Arachchilage, and Giovanni Russello. 2024. Ragent: Retrieval-based access control policy generation. *arXiv preprint arXiv:2409.07489* (2024).

[38] Zimo Ji, Daoyuan Wu, Wenyuan Jiang, Pingchuan Ma, Zongjie Li, Yudong Gao, Shuai Wang, and Yingjiu Li. 2026. Taming Various Privilege Escalation in LLM-Based Agent Systems: A Mandatory Access Control Framework. *arXiv preprint arXiv:2601.11893* (2026).

[39] Hebah I Abu Kaf, Khair Eddin Sabri, and Nadim Obeid. 2026. Temporal Defeasible Role–Group–Task Access Control: A Logic-Based Framework for Explainable Authorization. *IEEE Access* (2026).

[40] Subbarao Kambhampati, Karthik Valmeekam, Lin Guan, Mudit Verma, Kaya Stechly, Siddhant Bhambri, Lucas Paul Saldyt, and Anil B Murthy. 2024. Position: LLMs can't plan, but can help planning in LLM-modulo frameworks. In *Forty-first Int. Conf. on Machine Learning*.

[41] Đorđe Klisura, Joseph Khoury, Ashish Kundu, Ram Krishnan, and Anthony Rios. 2025. Role-Conditioned Refusals: Evaluating Access Control Reasoning in Large Language Models. *arXiv preprint arXiv:2510.07642* (2025).

[42] Dezhang Kong, Shi Lin, Zhenhua Xu, Zhebo Wang, Minghao Li, Yufeng Li, Yilun Zhang, Hujin Peng, Xiang Chen, Zeyang Sha, et al. 2025. A survey of llm-driven ai agent communication: Protocols, security risks, and defense countermeasures. *arXiv preprint arXiv:2506.19676* (2025).

[43] Ana Kovacevic and Dragana Nikolic. 2015. Cyber attacks on critical infrastructure: Review and challenges. *Handbook of research on digital crime, cyberspace security, and information assurance* (2015), 1–18.

[44] Ashutosh Kumar, Shiv Vignesh Murthy, Sagarika Singh, and Swathy Ragupathy. 2024. The ethics of interaction: Mitigating security threats in llms. *arXiv preprint arXiv:2401.12273* (2024).

[45] Martti Lehto. 2022. Cyber-attacks against critical infrastructure. In *Cyber security: Critical infrastructure protection*. Springer, 3–42.

[46] Qinbin Li, Junyuan Hong, Chulin Xie, Jeffrey Tan, Rachel Xin, Junyi Hou, Xavier Yin, Zhun Wang, Dan Hendrycks, Zhangyang Wang, et al. 2024. Llm-pbe: Assessing data privacy in large language models. *preprint arXiv:2408.12787* (2024).

[47] Yuetian Mao, Junjie He, and Chunyang Chen. 2025. From prompts to templates: A systematic prompt template analysis for real-world LLMapps. In *Proceedings of the 33rd ACM Int. Conf. on the Foundations of Software Engineering*. 75–86.

[48] Jennifer Mondragon, Gael Cruz, Carlos Rubio-Medrano, and Dvijesh Shastri. 2025. "I Apologize For Not Understanding Your Policy": Exploring the Evaluation of User-Managed Access Control Policies by AI Virtual Assistants. In *Proc. of the 2025 Workshop on Human-Centered AI Privacy and Security (HAIPS '25)*. 43–53.

[49] Jennifer Mondragon, Gael Cruz, Dvijesh Shastri, and Carlos E. Rubio-Medrano. 2024. Circles of Trust: A Voice-Based Authorization Scheme for Securing IoT Smart Homes. In *Proc. of the 29th ACM SACMAT* (San Antonio, TX, USA) *(SACMAT 2024)*. Association for Computing Machinery, New York, NY, USA, 193–195.

[50] Souradip Nath, Ananta Soneji, Jaejong Baek, Tiffany Bao, Adam Doupé, Carlos Rubio-Medrano, and Gail-Joon Ahn. 2025. "It's almost like Frankenstein": Investigating the Complexities of Scientific Collaboration and Privilege Management

within Research Computing Infrastructures . In *2025 IEEE Symp. on Security and Privacy (SP)*. 3236–3254.
[51] Souradip Nath, Ananta Soneji, Jaejong Baek, Carlos Rubio Medrano, and Gail-Joon Ahn. 2025. Towards Collaboration-Aware Resource Sharing in Research Computing Infrastructures. In *The 10th IEEE Int. Conf. on Collaboration and Internet Computing (CIC 2025)*. 10.
[52] National Institute of Standards and Technology. 2020. Role-Based Access Control—RBAC. https://csrc.nist.gov/projects/role-based-access-control#economic-impact. [Online; accessed May-4-2026].
[53] National Institute of Standards and Technology (NIST). 2025. CAESARS Framework Extension: An Enterprise Continuous Monitoring Technical Reference Architecture. https://csrc.nist.gov/pubs/ir/7756/2pd [Online; accessed May-4-2026].
[54] Pantaleone Nespoli, Dimitrios Papamartzivanos, Félix Gómez Mármol, and Georgios Kambourakis. 2017. Optimal countermeasures selection against cyber attacks: A comprehensive survey on reaction frameworks. *IEEE Communications Surveys & Tutorials* 20, 2 (2017), 1361–1396.
[55] Lin Ning, Luyang Liu, Jiaxing Wu, Neo Wu, Devora Berlowitz, Sushant Prakash, Bradley Green, Shawn O'Banion, and Jun Xie. 2025. User-LLM: Efficient LLM Contextualization with User Embeddings. In *Proc. of the ACM on Web Conf. 2025 (WWW '25)*. 1219–1223.
[56] Nitish Ojha and Abhishek Vaish. 2025. In *Zero-Trust Learning*. Apple Academic Press, 305–328.
[57] Abdul-Rasheed Olatunji Ottun and Huber Flores. 2025. Trustworthy AI in Practice: A Comprehensive Review of Human Oversight and Human-in-the-Loop Approaches. *Authorea Preprints* (2025).
[58] Sean Peisert, Von Welch, Andrew Adams, RuthAnne Bevier, Michael Dopheide, Rich LeDuc, Pascal Meunier, Steve Schwab, and Karen Stocks. 2017. Open Science Cyber Risk Profile (OSCRP). doi:10.5281/zenodo.7268749
[59] Francesco Periti and Soumadeep Saha. 2026. The Synergistic Integration of Access Control Management and Large Language Model Agents: A Survey. *Authorea Preprints* (2026).
[60] Luca Petrillo, Fabio Martinelli, Antonella Santone, and Francesco Mercaldo. 2025. An Explainable Method for Automatic Extraction of Natural Language Access Control Policy Key Components. *Applied Sciences* 15, 22 (2025), 11854.
[61] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. 2017. Robust adversarial reinforcement learning. In *International Conf. on machine learning*. PMLR, 2817–2826.
[62] Mian Qian, Abubakar Ahmad Musa, Milon Biswas, Yifan Guo, Weixian Liao, and Wei Yu. 2025. Survey of artificial intelligence model marketplace. *Future Internet* 17, 1 (2025), 35.
[63] rci-asu [n. d.]. RC - Arizona State University. https://cores.research.asu.edu/research-computing/about. Accessed: 2026-01-15.
[64] rci-ub [n. d.]. RCI—University at Buffalo. https://www.buffalo.edu/ccr/services/high-performance-computing.html. Accessed: 2025-05-28.
[65] rci-uch [n. d.]. RCI—Ohio State College of Medicine. https://medicine.osu.edu/research/research-information-technology/services-and-products/research-computing-infrastructure. Accessed: 2025-05-28.
[66] Hirad Rezaei, Amin Beheshti, and Fethi Rabhi. 2025. Training LLM with Human Feedback. In *Handbook of Human-Centered Artificial Intelligence*. Springer, 1–62.
[67] Neele Roch, Hannah Sievers, Lorin Schöni, and Verena Zimmermann. 2024. Navigating autonomy: unveiling security experts' perspectives on augmented intelligence in cybersecurity. In *20th Symp. on Usable Privacy and Security*. 41–60.
[68] Scott Rose, Oliver Borchert, Stuart Mitchell, and Sean Connelly. 2020. Zero Trust Architecture. doi:10.6028/NIST.SP.800-207 https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=930420 Special Publication (NIST SP), National Institute of Standards and Technology.
[69] Carlos E. Rubio-Medrano, Akash Kotak, Wenlu Wang, and Karsten Sohr. 2024. Pairing Human and Artificial Intelligence: Enforcing Access Control Policies with LLMs and Formal Specifications. In *Proc. of the 29th ACM SACMAT* (San Antonio, TX, USA). 105–116.
[70] Carlos E Rubio-Medrano, Josephine Lamp, Ziming Zhao, Adam Doupé, and Gail-Joon Ahn. 2017. Mutated Policies: Towards Proactive Attribute-based Defenses for Access Control. In *2017 Workshop on Moving Target Defense*. ACM.
[71] Carlos E. Rubio-Medrano, Ziming Zhao, Adam Doupe, and Gail-Joon Ahn. 2015. Federated Access Management for Collaborative Network Environments: Framework and Case Study. In *Proc. of the 20th ACM Symp. on Access Control Models and Technologies (SACMAT '15)*. 125–134.
[72] Zaynab Sai'd. 2025. Explainable AI (XAI) in Identity Access Management: Bridging Trust and Transparency in User Authentication. (2025).
[73] Mayra Samaniego and Ralph Deters. 2018. Zero-trust hierarchical management in IoT. In *2018 IEEE Int. congress on Internet of Things (ICIOT)*. IEEE, 88–95.
[74] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. 1996. Role-Based Access Control Models. *Computer* 29, 2 (Feb. 1996), 38–47.
[75] Debdeep Sanyal, Umakanta Maharana, Yash Sinha, Hong Ming Tan, Shirish Karande, Mohan Kankanhalli, and Murari Mandal. 2025. Orgaccess: A benchmark for role based access control in organization scale llms. *arXiv preprint arXiv:2505.19165* (2025).
[76] Atharva Shah, Ishani Mathur, and Harshil Kanakia. 2025. Understanding and Mitigating Data Poisoning in LLMs. In *2025 Int. Conf. on Responsible, Generative and Explainable AI (ResGenXAI)*. IEEE, 1–4.
[77] Aashish Sharma, Zbigniew Kalbarczyk, R Iyer, and James Barlow. 2010. Analysis of credential stealing attacks in an open networked environment. In *2010 fourth Int. Conf. on network and system security*. IEEE, 144–151.
[78] Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R Johnston, et al. 2023. Towards understanding sycophancy in language models. *arXiv preprint arXiv:2310.13548* (2023).
[79] Tianneng Shi, Jingxuan He, Zhun Wang, Hongwei Li, Linyu Wu, Wenbo Guo, and Dawn Song. 2025. Progent: Programmable privilege control for llm agents. *arXiv preprint arXiv:2504.11703* (2025).
[80] Chengyu Song, Linru Ma, Jianming Zheng, Jinzhi Liao, Hongyu Kuang, and Lin Yang. 2024. Audit-llm: Multi-agent collaboration for log-based insider threat detection. *arXiv preprint arXiv:2408.08902* (2024).
[81] Pratik Sonune, Ritwik Rai, Shamik Sural, Vijayalakshmi Atluri, and Ashish Kundu. 2025. LMN: A tool for generating machine enforceable policies from natural language access control rules using LLMs. *arXiv preprint arXiv:2502.12460* (2025).
[82] Nan Sun, Ming Ding, Jiaojiao Jiang, Weikang Xu, Xiaoxing Mo, Yonghang Tai, and Jun Zhang. 2023. Cyber threat intelligence mining for proactive cybersecurity defense: A survey and new perspectives. *IEEE Communications Surveys & Tutorials* 25, 3 (2023), 1748–1774.
[83] Yuxi Sun, Wenbo Shang, Wei Gao, Xin Huang, and Jing Ma. 2026. Insist when Know, Caution when Not Know? Unveiling LLMs' Fact-Checking Behavior Amidst Knowledge Conflicts. https://openreview.net/pdf?id=VgEEl1UDlg.
[84] Georgios Syros, Anshuman Suri, Jacob Ginesin, Cristina Nita-Rotaru, and Alina Oprea. 2025. Saga: A security architecture for governing ai agentic systems. *arXiv preprint arXiv:2504.21034* (2025).
[85] The Apache Foundation. 2026. Apache License, Version 2.0. http://www.apache.org/licenses/LICENSE-2.0 [Last Visited: January 13, 2026].
[86] Lillian Tsai and Eugene Bagdasarian. 2025. Contextual agent security: A policy for every purpose. In *Proc. of Workshop on Hot Topics in Operating Systems*. 8–17.
[87] Yu-Min Tseng, Yu-Chao Huang, Teng-Yun Hsiao, Wei-Lin Chen, Chao-Wei Huang, Yu Meng, and Yun-Nung Chen. 2024. Two tales of persona in llms: A survey of role-playing and personalization. In *Findings of the Assoc. for Computational Linguistics: EMNLP 2024*. 16612–16631.
[88] Kun Wang, Guibin Zhang, Zhenhong Zhou, Jiahao Wu, Miao Yu, Shiqian Zhao, Chenlong Yin, Jinhu Fu, Yibo Yan, Hanjun Luo, et al. 2025. A comprehensive survey in llm (-agent) full stack safety: Data, training and deployment. *arXiv preprint arXiv:2504.15585* (2025).
[89] Wenbo Wang, Ying Zhang, Tao Xue, Haibin Zhang, and Yanbin Wang. 2026. NLACPs Dataset Expansion with AI: Leveraging ChatGPT for Policy Generation. *IEEE Internet of Things Journal* (2026).
[90] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data* 3, 1 (2016), 1–9.
[91] Yuhao Wu, Ke Yang, Franziska Roesner, Tadayoshi Kohno, Ning Zhang, and Umar Iqbal. 2025. Towards automating data access permissions in ai agents. *arXiv preprint arXiv:2511.17959* (2025).
[92] Mian Yang, Vijayalakshmi Atluri, Shamik Sural, and Ashish Kundu. 2025. Extraction of machine enforceable ABAC policies from natural language text using LLM knowledge distillation. In *Proc. of the 30th ACM Symp. on Access Control Models and Technologies*. 157–168.
[93] Mingjie Yu, Fenghua Li, Nenghai Yu, Xiao Wang, and Yunchuan Guo. 2022. Detecting conflict of heterogeneous access control policies. *Digital Communications and Networks* 8, 5 (2022), 664–679.
[94] Kaiyuan Zhang, Zian Su, Pin-Yu Chen, Elisa Bertino, Xiangyu Zhang, and Ninghui Li. 2025. LLM Agents Should Employ Security Principles. *arXiv preprint arXiv:2505.24019* (2025).
[95] Zhehao Zhang, Ryan A Rossi, Branislav Kveton, Yijia Shao, Diyi Yang, Hamed Zamani, Franck Dernoncourt, Joe Barrow, Tong Yu, Sungchul Kim, et al. 2024. Personalization of large language models: A survey. *arXiv preprint arXiv2411.00027* (2024).
[96] Peter Yong Zhong, Siyuan Chen, Ruiqi Wang, McKenna McCall, Ben L Titzer, Heather Miller, and Phillip B Gibbons. 2025. Rtbas: Defending llm agents against prompt injection and privacy leakage. *arXiv preprint arXiv:2502.08966* (2025).
[97] Jinhao Zhu, Kevin Tseng, Gil Vernik, Xiao Huang, Shishir G Patil, Vivian Fang, and Raluca Ada Popa. 2025. MiniScope: A Least Privilege Framework for Authorizing Tool Calling Agents. *arXiv preprint arXiv:2512.11147* (2025).
[98] Sharif Noor Zisad and Ragib Hasan. 2026. LLMAC: A Global and Explainable Access Control Framework with Large Language Model. In *2026 IEEE 23rd Consumer Communications & Networking Conf. (CCNC)*. IEEE, 1–6.