Slide 4: Contexto "Agora, vamos mergulhar no contexto do nosso trabalho. A arquitetura RAG é, em teoria, uma ferramenta para tornar os LLMs mais seguros e factuais em ambientes corporativos. No entanto, na prática, ela cria uma nova e perigosa superfície de ataque. Vulnerabilidades como

prompt injection e **envenenamento de documentos** se tornam particularmente eficazes nesse cenário.

Quando um ataque desses é bem-sucedido, as consequências são severas, indo desde o vazamento de dados corporativos e pessoais até a geração de desinformação estratégica, o que pode comprometer decisões de negócio. Por essa razão, nosso trabalho se dirige diretamente às lideranças executivas e às equipes de TI e segurança da informação, que estão na linha de frente, responsáveis por implementar e, crucialmente, proteger esses sistemas complexos."

Slide 5: Motivação "Isso nos conduz diretamente à nossa motivação. As soluções de segurança atuais são insuficientes para lidar com a complexidade desses novos ataques. Ferramentas genéricas, conhecidas como 'guardrails', geralmente operam com base em filtros de palavras-chave ou classificadores simples. Elas falham em detectar ataques mais sofisticados, como instruções maliciosas disfarçadas de texto comum ou comandos ofuscados.

Além disso, os detectores de 'jailbreak', que buscam por padrões de ataque conhecidos, podem ser facilmente enganados por pequenas variações. O ponto mais crítico em um sistema RAG é que um comando malicioso injetado em um documento recuperado pode efetivamente suprimir as diretrizes de segurança originais do modelo, fazendo com que ele ignore suas próprias regras. Essa fragilidade mostra que soluções isoladas não oferecem a proteção robusta que um ambiente empresarial exige."

Slide 6: Justificativa "E é aqui que a nossa justificativa se fundamenta. Acreditamos que a única abordagem verdadeiramente eficaz é uma estratégia de

defesa em profundidade. Em vez de confiar em uma única barreira de segurança, que pode ser contornada, nós propomos uma arquitetura que integra múltiplas camadas defensivas que operam em sinergia.

Isso inclui, por exemplo, a implementação de

filtros semânticos para analisar e classificar o risco dos documentos *antes* que o LLM os processe ; a

validação da saída gerada pelo modelo para identificar padrões suspeitos antes de exibi-los ao usuário ; e o

isolamento de ferramentas externas em ambientes controlados, como sandboxes, para limitar o dano potencial de uma ação maliciosa. Estudos recentes mostram que os ataques mais bem-sucedidos exploram falhas em múltiplas fases do pipeline, o que ressalta a necessidade de uma defesa unificada e sólida."

Slide 7: Objetivos "Com base nesse cenário, nosso objetivo principal é resolver a questão crítica da segurança em modelos que utilizam a arquitetura RAG. Para alcançar isso, nosso trabalho se concentra em desenvolver e implementar novos e robustos mecanismos de defesa, projetados especificamente para detectar e neutralizar as ameaças que discutimos. O resultado final que buscamos é um sistema que não apenas funcione bem, mas que garanta a segurança e a confiabilidade necessárias para proteger tanto os dados da empresa quanto os de seus usuários."

Slide 9: Artigos de Periódicos (Seção Expandida) "Para fundamentar nossa proposta, aprofundamos em três artigos de periódicos Qualis A1 que são pilares para a nossa pesquisa.

O primeiro artigo, publicado na

IEEE Transactions on Information Forensics and Security. Ele investiga o uso de LLMs para gerar automaticamente afirmações de segurança para projetos de hardware, uma tarefa extremamente delicada. A grande conclusão do estudo é que, embora os LLMs possam realizar a tarefa, eles possuem limitações significativas e o conteúdo que produzem precisa de validação e supervisão rigorosas para ser confiável. Para o nosso projeto, essa conclusão é vital: ela prova que não se pode confiar cegamente na saída de um LLM, mesmo quando a tarefa é de segurança. Isso justifica diretamente a necessidade de uma camada de validação em nossa arquitetura de defesa em profundidade.

O segundo artigo, da revista

COMPUTERS & SECURITY, apresenta uma ferramenta inovadora chamada SLFHunter para *'Detectar vulnerabilidades de injeção de comandos em firmware embarcado baseado em Linux com análise de contaminação baseada em LLM'*. O SLFHunter combina análise de fluxo de dados com prompts inteligentes para que um LLM identifique vulnerabilidades que ferramentas tradicionais não conseguem detectar. Ele nos mostra que podemos integrar a própria IA como um mecanismo de defesa ativo em nossa arquitetura, por exemplo, usando um LLM para analisar e filtrar documentos em busca de ameaças ocultas.

Finalmente, o terceiro artigo, da

IEEE Transactions on Dependable and Secure Computing, apresenta o *'PrivacyAsst:* Este artigo aborda um dos maiores riscos dos sistemas integrados: o vazamento de dados privados quando o LLM utiliza ferramentas externas. O PrivacyAsst é um sistema que monitora ativamente as interações entre o modelo e os serviços conectados, conseguindo detectar e impedir vazamentos de dados sensíveis com um impacto mínimo no tempo de resposta. Este estudo é fundamental para nós, pois oferece um exemplo prático e eficaz de uma das camadas de defesa que propomos: o controle e monitoramento de ferramentas externas, provando que mecanismos de defesa proativos são essenciais para a IA em ambientes empresariais."