Propuesta de Tesina

13 de noviembre de 2024

Postulante: Tomás Castro Rojas
Director: Gustavo Betarte
Codirector: Carlos Luna

1. Situación del postulante

Al día 13/11/2024, el estudiante tiene pendiente rendir Ingeniería de Software II y una materia Optativa (Programación Verificada con F*). El resto de materias están aprobadas. Actualmente se encuentra trabajando para una empresa de software con dedicación de 30 horas semanales y se desempeña como docente auxiliar con dos cargos de dedicatoria simple.

2. Título

Formalización del comportamiento de atacantes usando la taxonomía MITRE ATT&CK

3. Motivación y Objetivo General

La capacidad de cómputo de los sistemas informáticos ha crecido de manera exponencial en la última década. No solo por los avances tecnológicos en hardware sino, y más importante, por la escala de los centros de datos (data centers) donde son alojados dichos sistemas. Para citar un ejemplo, en la última década Amazon Web Services cuadriplicó sus data centers, de 28 a 105 zonas de disponibilidad, y en consecuencia ha visto sus ganancias aumentar significativamente, de 4,5 mil millones USD a 90 mil millones de dólares [7, 11, 10]. Esto responde a un incremento en la cantidad de usuarios, es decir, a un aumento en la demanda de procesamiento. Sin embargo, el crecimiento en el tráfico de datos trae consigo un interés de agentes maliciosos por obtener información sensible o disrumpir servicios, generalmente en pos de un objetivo económico.

Es de vital importancia asegurar que las distintas operaciones del sistemas cumplen con las políticas de seguridad establecidas por la empresa u organización gubernamental y que no resulten en vulnerabilidades que comprometan la integridad y confidencialidad de los datos. Esto se logra habitualmente implementando mecanismos de defensa y/o mitigación de daños siguiendo técnicas y requerimientos de desarrollo estandarizados como ISO 27001, o verificando formalmente que el programa cumple con la especificación del modelo de seguridad del sistema.

Lamentablemente, no existen balas de plata para el desarrollo de software. La Ciberinteligencia ha ganado notoriedad como enfoque complementario a la hora de diseñar la seguridad de un sistema. Esta misma se centra en comprender el perfil del atacante, intentando conocer el comportamiento y evolución de su conocimiento a medida que progresa en su ataque a un sistema, y de este modo realizar una detección temprana de posibles amenazas.

El objetivo de este trabajo es realizar una verificación formal de un modelo de atacante propuesto en el experimento PWNJUTSU [4] en el asistente de pruebas Coq [9]. Estudiar formalmente un modelo verificado del comportamiento de atacantes es fundamental para mejorar nuestro entendimiento de los posibles vectores de ataque a un sistema. Principalmente permite demostrar propiedades relevantes del modelo que comprueban la robustez y capacidad de predictibilidad del enfoque.

4. Fundamentos y estado de conocimiento sobre el tema

Una amenaza persistente avanzada (APT, Advanced Persistent Threat) es un ciberataque encubierto contra una red informática en el que el atacante obtiene y mantiene un acceso no autorizado a la red objetivo y permanece sin ser detectado durante un periodo de tiempo significativo [2]. En años más recientes, organizaciones criminales han desarrollado herramientas con capacidad para lanzar campañas de ataque de características APT. Por este motivo, los sistemas informáticos deben tomar medidas para detectar comportamiento malicioso que comprometa la seguridad y el funcionamiento. El Centro de Tecnologías Emergentes de la Universidad Carnegie Mellon, define como uno de los aspectos fundamentales de la Ciberinteligencia "La adquisición y el análisis de información para identificar, rastrear y predecir las capacidades, intenciones y actividades cibernéticas que apoye la toma de decisiones" [1]. Esta disciplina pone el foco en el modus operandi de los atacantes para diseñar medidas de seguridad.

El entendimiento de secuencias de técnicas de ataque y el contexto donde ocurren mejora la eficacia defensiva de un sistema [3]. Con este objetivo, desde 2013 se trabaja activamente en el desarrollo colectivo del marco MITRE ATT&CK, una base de conocimiento de las Técnicas, Tácticas y Procedimientos (TTP) que utilizan los adversarios en sus campañas de ataque [8]. Este framework permite una descripción de las distintas fases del ciclo de vida de los ataques de los adversarios y las plataformas a las que se dirigen como objetivo. Además provee un lenguaje común para la comunicación de amenazas entre expertos de seguridad incentivando la colaboración en la prevención de las mismas.

En esta línea, el experimento PWNJUTSU propone un modelo de atacante que usa el framework ATT&CK. Este modelo describe el estado del atacante como su área de propagación, los secretos obtenidos y el conocimiento del entorno. El estado cambia a medida que el atacante ejecuta técnicas ATT&CK, que dada la condición del atacante puede resultar en la obtención de un secreto, mejorar su conocimiento del entorno o escalar privilegios dentro del sistema. Visto matemáticamente, PWNJUTSU define una maquina de estado abstracta con una semántica operacional expresada en técnicas ATT&CK.

El estudio de este modelo de atacante se hará desde una perspectiva formal, desarrollando una especificación en el Cálculo de Construcciones Inductivas [6], usando Coq [5]. La decisión de realizar un modelo formal se basa en las ventajas que otorga: poder definir

cuáles son capacidades del atacante sin ambigüedades y poder probar propiedades de seguridad, de manera computable, sobre la interacción del atacante con el sistema. Más aún, la especificación formal ofrece extensionalidad en el modelo ya que si el atacante desarrollara nuevas técnicas es posible probar de manera casi automática si las propiedades de seguridad siguen vigentes. Dado el enfoque en Ciberinteligencia, a partir de la especificación formal se espera poder probar la predictibilidad del modelo en la detección temprana de amenazas analizando si existen trazas de ejecución de técnicas que ocasionan una vulnerabilidad en un sistema.

5. Objetivos específicos

1. Desarrollar una especificación formal de un modelo de atacante según el experimento PWNJUTSU
2. Verificación formal del modelo especificado, usando el asistente de pruebas Coq. En particular, se prevé analizar propiedades relevantes de seguridad
3. Derivación de un prototipo funcional certificado del modelo (dependiendo del tipo de formalización que se haga)
4. Analizar, en base al trabajo previo, la robustez y capacidad de predictibilidad del enfoque

6. Metodología y Plan de Trabajo

1. Para alcanzar el objetivo específico 1 se dispondrá el material de base a formalizar. Asimismo, se utilizará como referencia al menos una formalización previa en otro dominio. A través de reuniones periódicas con los tutores del trabajo se discutirán elementos a considerar en la formalización.
   - especificación formal: 8 semanas
2. Para el segundo objetivo específico se discutirán propiedades de interés a analizar (formalmente)
   - análisis de propiedades: 8 semanas
3. Dependiendo del tipo de formalización desarrollada es posible extraer un prototipo funcional certificado del modelo usando el asistente de pruebas Coq. Se analizará en conjunto la viabilidad de obtener tal prototipo (no es un requisito estricto)
   - prototipo certificado: 3 semanas
4. Finalmente, en interacción con los tutores se espera poder realizar un análisis sobre la robustez y la capacidad de predictibilidad del enfoque
   - análisis: 3 semanas
   - INFORME FINAL: 6 semanas

Referencias

[1] Scott Ainslie, Dean Thompson, Sean Maynard y Atif Ahmad. "Cyber-threat intelligence for security decision-making: A review and research agenda for practice". En: *Computers Security* 132 (2023), pág. 103352. ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2023.103352.

[2] Adel Alshamrani, Sowmya Myneni, Ankur Chowdhary y Dijiang Huang. "A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities". En: *IEEE Communications Surveys Tutorials* 21.2 (2019), págs. 1851-1877. DOI: 10.1109/COMST.2019.2891891.

[3] Andy Applebaum, Desiree Beck, Mark E. Haase y Jon Baker. *Attack Flow*. MITRE Engenuity. 2022. URL: https://medium.com/mitre-engenuity/attack-flow-beyond-atomic-behaviors-c646675cc793 (visitado 10-2024).

[4] Aimad Berady, Mathieu Jaume, Valérie Viet Triem Tong y Gilles Guette. "PWNJUTSU: A dataset and a semantics-driven approach to retrace attack campaigns". En: *IEEE Transactions on Network and Service Management* (jun. de 2022). DOI: 10.1109/TNSM.2022.3183476.

[5] Yves Bertot, Pierre Castéran, Gérard (informaticien) Huet y Christine Paulin-Mohring. *Interactive theorem proving and program development: Coq'Art : the calculus of inductive constructions*. Texts in theoretical computer science. Données complémentaires http://coq.inria.fr. Berlin, New York: Springer, 2004. ISBN: 978-3-540-20854-9. URL: http://opac.inria.fr/record=b1101046.

[6] T. Coquand y G. Huet. "The Calculus of Constructions". En: *Information and Computation*. Vol. 76. 2/3. Academic Press, feb. de 1988, págs. 95-120.

[7] Lee Mathews. *AWS Data centers in 2014*. 2014. URL: https://web.archive.org/web/20191223045710/https://www.geek.com/chips/just-how-big-is-amazons-aws-business-hint-its-absolutely-massive-1610221/ (visitado 10-2024).

[8] *MITRE ATT&CK Matrix for Enterprise*. MITRE Engenuity. 2023. URL: https://attack.mitre.org/matrices/enterprise/ (visitado 10-2024).

[9] The Coq Development Team. *The Coq Proof Assistant Reference Manual*. Ver. 8.20. INRIA. URL: https://coq.inria.fr/ (visitado 10-2024).

[10] Lionel Sujay Vailshery. *AWS Revenue from 2013 to 2023*. URL: https://www.statista.com/statistics/233725/development-of-amazon-web-services-revenue/ (visitado 10-2024).

[11] Mary Zhang. *AWS Data centers in 2024*. URL: https://dgtlinfra.com/amazon-web-services-aws-data-center-locations/ (visitado 10-2024).