LEVERAGING BAYES THEOREM FOR ENHANCED AUTOMATED RED-TEAM OPERATIONS USING MITRE CALDERA

Members:
Reyes Lee Yui Hou
(Anglo-Chinese School (Independent))
Goh Shang Yu (Victoria School)

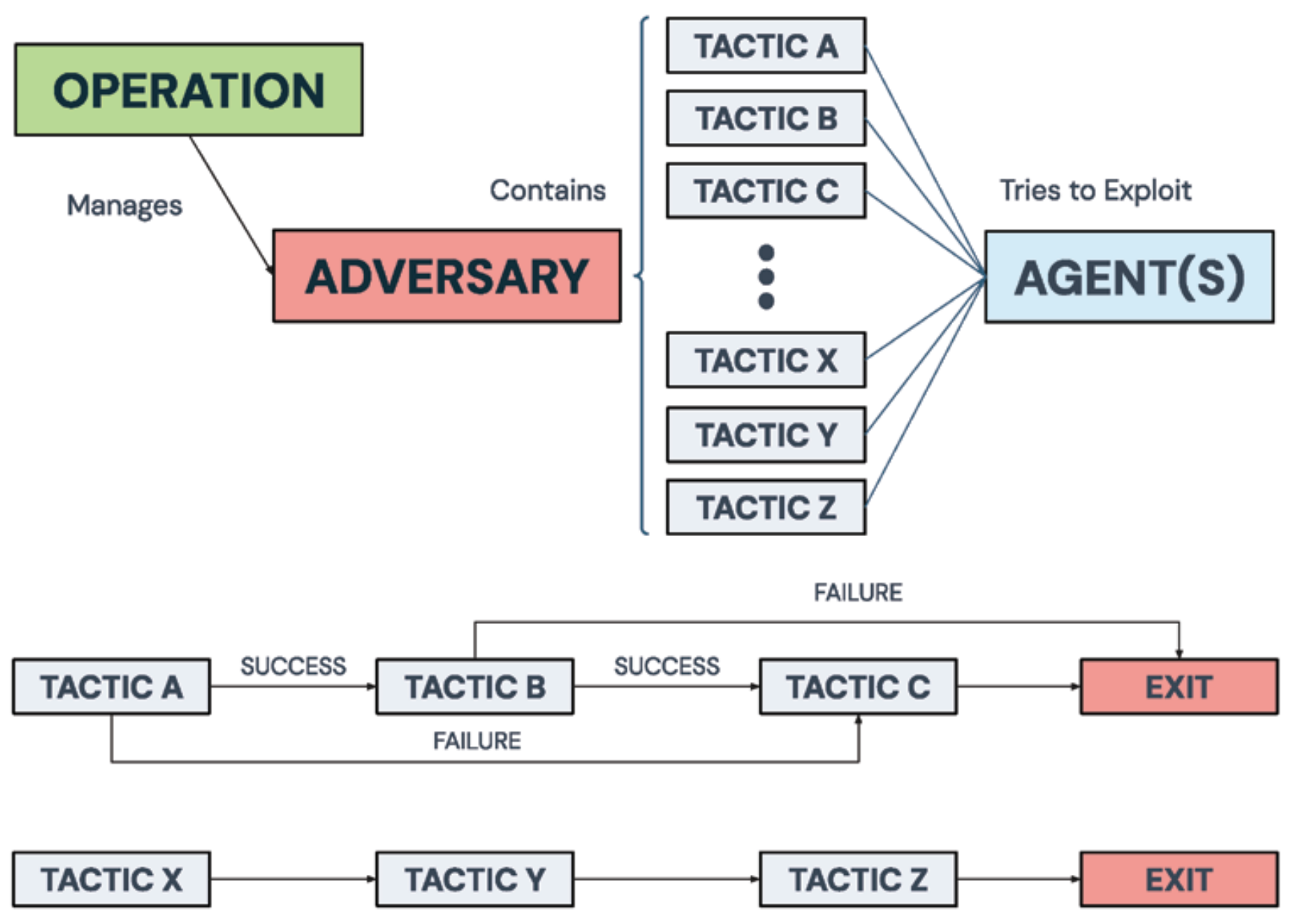
Mentor:
Lim Seh Leng
(Defence Science and Technology Agency)

Introduction

Red-teaming is a cybersecurity process for identifying vulnerabilities in computer systems by simulating cyberattacks. **MITRE Caldera** is an industry tool for automated adversary emulation, allowing users to build a threat profile using real-world Tactics, Techniques and Procedures (TTPs) from the MITRE ATT&CK Framework, and launch it on target systems.

The **planner** chosen at the start of the operation contains logic which determines the order in which abilities will be executed. Two planners of interest will be compared - the **atomic planner** and the **naive bayes planner**.

The atomic planner executes abilities in a predefined order, one by one, for each agent.



The bayes planner applies Bayes Theorem to prioritise attacks with higher probabilities of success based on past data and hence requires sufficient historical operational data. Bayes Theorem states the probability P of a hypothesis H given an evidence E is

$$P(H|E) = \frac{P(E|H) \times P(H)}{P(E|H) \times P(H) + P(E|\neg H) \times P(\neg H)} \quad (1)$$

The planner uses Equation 1 to update the expected probability of success for each potential action using each new piece of operational data.

The research will investigate the **significance of the planner** on an attack's effectiveness, **Bayes Theorem's applicability** in enhancing and streamlining the automated process, and **demonstrate** that algorithmic/conditional planners can optimise penetration testing.

Methodology

Dependent Variable	Method of Determination
Success Rate / %	Percentage of abilities that achieved successful execution without erroring out, timing out, or being denied by the target. Calculated using Equation 2. $\frac{\text{number of successful abilities}}{\text{total abilities executed}} \quad (2)$
Stealth	Total number of detected actions by antivirus software across all target computers. Obtained by manually checking Windows Defender logs after the entire operation is completed, and cross-verifying timestamps.

Procedure

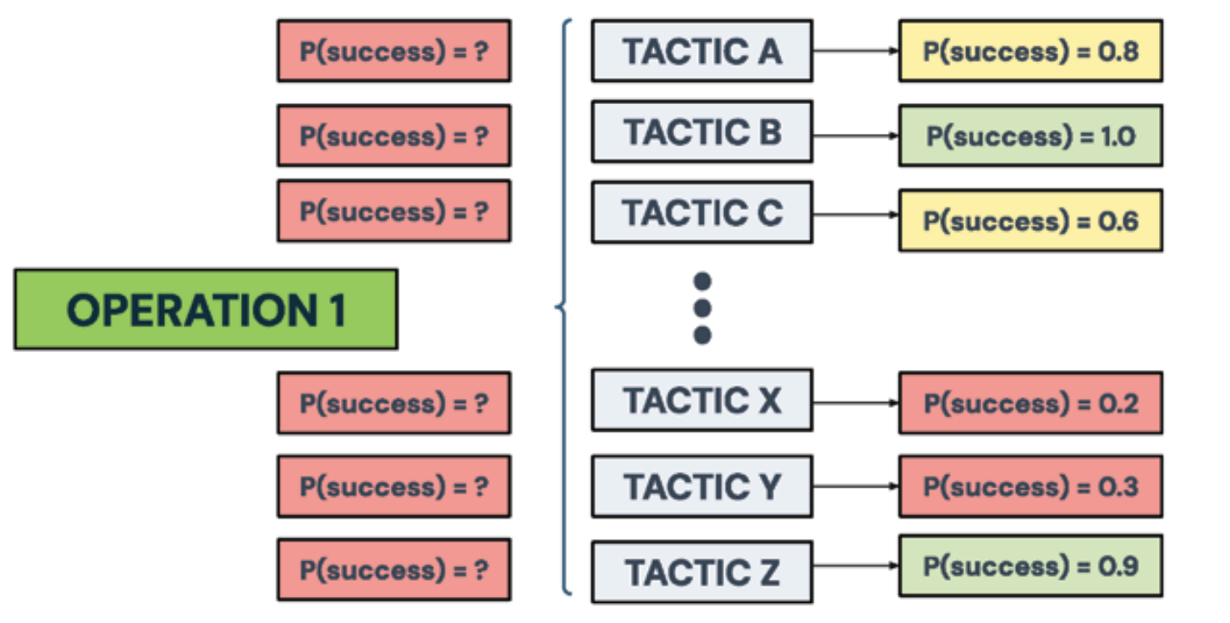
1. Three virtual machines running Windows 10 are set up on Oracle VirtualBox. Each machine is configured as a home-use computer, with files, passwords and apps. All software is checked to be running the same version.
2. Windows Defender is set to whitelist the agent.
3. A snapshot of each virtual machine is taken after setup is completed to ensure the same starting parameters.
4. An agent is launched on every virtual machine.
5. From the Caldera web console, an operation using a custom adversary profile is launched on every agent, with the atomic planner managing the operation.
6. The total number of abilities executed, the number of successful abilities and the number of antivirus warnings logged is recorded and tabulated. The planner's logs, which detail the planner's decision making, are also collected from Caldera.
7. Each virtual machine is reset to the original snapshot.
8. Steps 5-7 are repeated for a total of three iterations.
9. Steps 5-8 are repeated with the bayes planner.

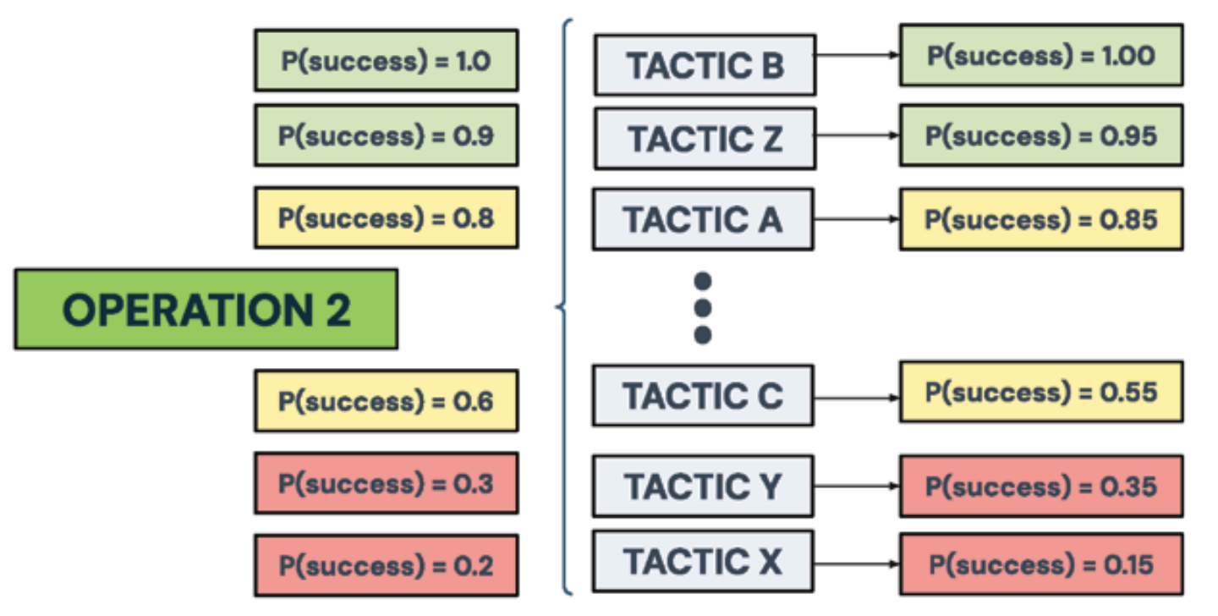
Results

Dependent Variable	Atomic Planner	Bayes Planner
Average Success Rate / %	$\frac{33+35+33}{3\times40} = 84.2\%$	$\frac{34+36+36}{3\times40} = 88.3\%$
Stealth / Total No. of Warnings	$6+8+5=19$	$7+6+5=18$

The increase in success rate of the bayes planner can be attributed to it making decisions to avoid executing certain abilities that had too low a chance of success, as determined by past operational data. In contrast, the atomic planner was programmed to execute every ability even if it had a significant known history of failure. This wastes time and resources when running an automated attack.

The slight improvement in stealth can be explained similarly. If certain abilities frequently triggered antivirus system in prior attempts, the bayes planner skips those abilities, improving its stealth profile.





The results can be visualised better with secondary data collected from the planner's logs. In the first figure, the bayes planner begins with the atomic ordering (e.g. A-Z) as there is no data available. As the operation completes, the planner mathematically learns which abilities have the highest prospects of success.

The second figure shows prioritisation of tactics most likely to succeed, based on the historical success rate collected in the previous operations. If abilities do not satisfy the minimum probability of success threshold setting, they may not be executed. The increasing precision of P(success) reflects the planner's decision-making capabilities continually refining itself as more data becomes available.

Hence, the conditions required to maximise operational efficiency and success rate are a **large amount of historical data** and an **optimised minimum success threshold setting**.

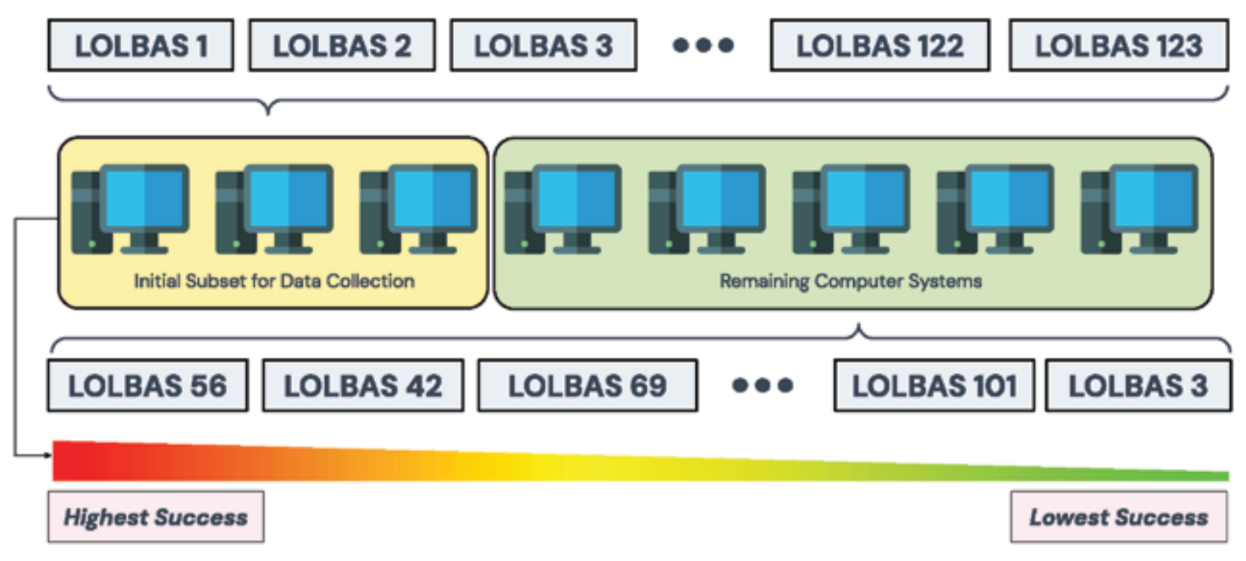
Conclusion

The research concludes that the **sequence and manner** in which an automated red-team attack is executed is **highly important**. Execution strategy impacts effectiveness; the **size of that impact depends on the scale** of the operation.

The bayes planner's ability to adapt and optimise attack performance based on historical data demonstrates that **Bayes Theorem can be applied to enhance adversary emulation**. Optimal operational efficiency of the bayes planner is achieved when a **large amount of historical operational data** is available and the **parameters of the planner** are favourably tuned.

Algorithmic/Conditional planners can streamline operations by helping testers **identify attacks with the highest probability of success**, conserving resources and allowing more efficient adversary emulation.

Real-World Applications



When testing large numbers of Windows systems for Living Off the Land Binaries and Scripts (LOLBAS), which are trusted native binaries that attackers can abuse, testing all possible LOLBAS on all systems would be extremely inefficient and resource-intensive.

The bayes planner can streamline this by testing a subset of systems to identify ineffective LOLBins and focus on high-impact techniques. This improves testing coverage in less time, enabling quicker remediation of vulnerabilities. The bayes planner can **increase the productivity** of cybersecurity testers by making it much **easier to pinpoint the best starting point** of attacks.