For security audit trails analysis as an alternative tool

CasSata, A Generic Algorithm
A heuristic mechanism (genetic algorithm)

A pessimistic approach

No timing aspect in attack scenario

To investigate misuse detection

**CSSATA: Main Ideas**
A pessimistic approach ↓

Max

Maximum = H

Our View of the Security Audit Trail Analysis
A tool: a genetic algorithm

This process is repeated until a solution is found

better hypotheses

According to this evaluation, derivation of a new (and

Hypotheses assessment

Hypotheses is made

Hypotheses approach:

A heuristic approach

systematic exploration impossible

possible values equivalent $2^{-N}$
A Simple Genetic Algorithm
Two-part fitness function

Individual

Maximum

Individuals and Fitness Function
the base?

How does it evolve in function of the number of attacks in

- Is the running time satisfactory?

- What is the final population?

- How does the population evolve?

Questions:

- Sequences of commands on networks are included in the audit vectors generated from the

- The attack base contains between 24 and 200 attacks

- (attack)

- Users: sequences of commands over a 30 minute period (no

- Data generated by the AIX audit sub-system

Experiments
We know that only performed attack I was.

\[
\begin{array}{cccc}
0 & 0 & 1 & 0 \\
0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 \\
1 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
\end{array}
\]

\[
\frac{8}{9} = \frac{\text{true}}{\text{false}} \quad \frac{9}{d} = \frac{\text{false}}{\text{true}}
\]

\(0 = ^pL\) ideally to absent attacks equal 1 out of the total number of individuals in which this corresponding number of individuals in which this corresponding

\(1 = ^dL\) ideally to present attacks equal 1 out of the total number of individuals in which this corresponding number of individuals in which this corresponding

\(\frac{^pL}{^dL}\) and \(\frac{^dL}{^pL}\)

**Define the Rations and** How to Evaluate the Quality of the Results?
The number of attacks actually present in the trial have no present and absent attacks. A good discrimination between $\Lambda$ and $\Lambda^a$.
Running Time

The duration of the audit session has no influence on the

The running time does not grow exponentially

IBM RS6000 320

28 types of events in the matrix

γ generations for constant L and L

ρ = 0.7, 7 μ = 0.002, 200 individuals

<table>
<thead>
<tr>
<th>Rate</th>
<th>Time</th>
<th>Attacks</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.3 × 10⁻⁹</td>
<td>625™</td>
<td>200</td>
</tr>
<tr>
<td>5.9 × 10⁻⁹</td>
<td>104™</td>
<td>100</td>
</tr>
<tr>
<td>5 × 10⁻⁸</td>
<td>32™</td>
<td>40</td>
</tr>
<tr>
<td>3 × 10⁻⁷</td>
<td>18™</td>
<td>24</td>
</tr>
</tbody>
</table>

Execution Time vs Number of Attacks in the Base
Find a competitive measurement process

- Improve our attack base
  (be rewritten)

- Use CSSA in a real environment (some code should

Future work:

- We do not precisely locate attacks in the audit trail
- We cannot detect the multiple realization of a particular

What we do not do:

Conclusion