Confidentiality-Preserving Refinement

Roberto Guanciale
Christoph Baumann
Mads Dam
Hamed Nemati

Entropy 16/06/2019
Refinement

- Change data representation
  - From C to assembly
  - From ideal function to SMC
- Add details
  - Caches
  - Timing
  - Addresses of variables
- Remove non-determinism
- Goal: prevent unintended leakage of secret data
Challenges

- Simple accounts of refinement (e.g., trace inclusion) do not guarantee confidentiality properties
- Several ways to specify licit information flows
  - multi-level security, decentralised model
  - declassification
  - ...
- Abstract model “specifies” the intended information flows

```plaintext
IF input = master-pwd
   output = MAC(key, data)
ELSE
   output = NULL
```
Challenges

- Simple accounts of refinement (e.g., trace inclusion) do not guarantee confidentiality properties
- Several ways to specify licit information flows
  - multi-level security, decentralised model
  - declassification
  - ...
- Abstract model “specifies” the intendend information flows

```plaintext
IF input = master-pwd
  output = MAC(key, data)
ELSE
  output = NULL
```

master-pwd can affect execution time of comparison

key can affect cache state due to table look-up
Observation equivalence

- Same attacker’s observations
  - I.e. input, data, output
Knowledge

\[ K(s, n) = \{ s_1 \mid \forall n_1 \leq n. s \downarrow_{n_1} \sim s_1 \downarrow_{n_1} \} \]
Knowledge

\[ K(s, n) = \{ s_1 \mid \forall n_1 \leq n. s \downarrow_{n_1} \sim s_1 \downarrow_{n_1} \} \]

- s.input = s.m-pwd iff s1.input = s1.m-pwd
Knowledge

\[ K(s, n) = \{ s_1 \mid \forall n_1 \leq n . s \downarrow_{n_1} \sim s_1 \downarrow_{n_1} \} \]

- \( s \text{ input} = s \text{ m-pwd} \iff s_1 \text{ input} = s_1 \text{ m-pwd} \)

- if \( s \text{ input} = s \text{ m-pwd} \) then
  - \( S(s \text{ key}, s \text{ data}) = S(s_1 \text{ key}, s_1 \text{ data}) \)

- Yardstick for information flows
Confidentiality Preserving Refinement \[ K(\bar{s}, n) = K([\bar{s}], n) \]
Confidentiality Preserving Refinement \[ K(\bar{s}, n) = K([\bar{s}], n) \]
Confidentiality Preserving Refinement $[K(\bar{s}, n)] = K([\bar{s}], n)$
Confidentiality Preserving Refinement $[K(\bar{s}, n)] = K([\bar{s}], n)$
Confidentiality Preserving Refinement $[K(\bar{s}, n)] = K([\bar{s}], n)$
Behavioral morphing refinements

- Attacker behavior could diverge due to low-level features (row-hammer, mismatched cacheability, weak memory models)
Behavioral morphing refinements
Behavioral morphing refinements

\[
\begin{align*}
\text{IF } \text{input} &= \text{master-pwd} \text{ TRUE} \\
\text{output} &= \text{MAC(key, data)} \\
\text{ELSE} \\
\text{output} &= \text{NULL}
\end{align*}
\]
Thank You