Towards a Notion of Quantitative Security Analysis

Iliano Cervesato iliano@math.tulane.edu

Tulane University – New Orleans, LA

http://theory.stanford.edu/~iliano/

Qualitative (Dolev-Yao) Analysis

- Classifies protocol operations in
  - Possible (Dolev-Yao)
    - Reception/transmission
    - Crypto with key, …

  "Easy" (polynomial)
  - Impossible
    - Guessing keys
    - Breaking crypto, …

  "Hard" (exponential)
- Security assessed <u>only</u> on possible ops
  - "Easily" achieved by most current tools
  - What next?

Analysis beyond Dolev-Yao

Data

Symbolic → Bit-oriented

Crypto

Perfect ↓ Real

More ops
- xor
- DH, …

Type confusion

Guessing

Probabilistic

Cost-aware

Crypto hybrid
- probability
- complexity

Cost-Aware Security Analysis

- Assign cost to operations [Meadows,01]
  - Including non Dolev-Yao
    - Discrete logarithm, factoring, …
    - (Verifiable) guessing [Lowe,02]
    - Principal subversion, …
- Applications
  - Estimate actual resources needed for attacks
  - Resources limitation (smart cards, PDAs, …)
  - DoS resistance assessment
  - Comparing attacks or protocols

Protocol Example

$A \to B: \{n_A, A\}_{kB}$
$B \to A: \{n_A, n_B\}_{kA}$
$A \to B: \{n_B\}_{kB}$

- Needham-Schroeder protocol
  - Initiator role

$$\begin{bmatrix} PrvK_A(k_A, k'_A), \\ PubK_*(B, k_B) \end{bmatrix} \rightarrow \exists n_A. \begin{bmatrix} PrvK_A(k_A, k'_A), \\ PubK_*(B, k_B), \\ L(k_A, k'_A, k_B, n_A), \\ N(\{n_A, A\}_{kB}) \end{bmatrix}$$

$$\begin{bmatrix} L(k_A, k'_A, k_B, n_A), \\ N(\{n_A, n_B\}_{kA}) \end{bmatrix} \rightarrow \begin{bmatrix} N(\{n_B\}_{kB}) \end{bmatrix}$$

Traces and Scripts

- Traces
  - Rewrite sequence $(r_1,\theta_1),\ldots,(r_n,\theta_n)$ from M_0 to M_n
    - Rules r_i
    - Substitutions θ_i
- Scripts
  - Parametric traces
    - S, (r,ξ)
    - $S_1 + S_2$
    - $!_n\ S$
  - Normal run: S_{NR}
  - Attack scripts: S_A

Preparing for Cost Assignment

- Isolate operations
  - Verification
    - Success
    - Failure
  - Construction

  - Split LHS into atomic steps
  - Allow failure
- Apply rule in stages
  - Pre-screening
  - Detailed verification

Fine-Grained MSR

- Verification rules
  - $N^h(x) \rightarrow R(x)$
  - $\mathbf{R(y)}, R'(op_{\underline{y}}(\underline{\mathbf{x}})) \rightarrow \mathbf{R''(x)}$ else cr
  - $R(x), R'(x) \rightarrow .$ else cr
  - …
- Construction rules
  - Remain the same

Fine-Grained Intruder

$I(g), I(g^x) \rightarrow I(x)$

Dolev-Yao style

- $N^h(x) \rightarrow I(x)$
- $M^*(x) \rightarrow I(x)$
- $\mathbf{I(y)}, I(op_{\underline{y}}(\underline{x})) \rightarrow \mathbf{I(x)}$
- $I(x) \rightarrow N^h(x)$
- $. \rightarrow \exists x.\ I(x)$
- $\mathbf{I(x)} \rightarrow I(op(\underline{x}))$

Subversion

- $. \rightarrow X(A)$
- $X(A) \rightarrow .$
- $X(A), M_A(x) \rightarrow X(A), \mathbf{I(x)}$

Guessing

$\ldots \rightarrow G(x)$

$\ldots \rightarrow V_1(m_1)$

$\ldots \rightarrow V_2(m_2)$

$G(x), V_1(y), V_2(y) \rightarrow I(x)$

Cost

$$\sum v\tau^{A}$$

- τ: cost type
  - Time, space, energy, …
- A: principal incurring cost
- v: amount of cost
  - Physical measurements
  - 0 / ∞ (Dolev-Yao model)
  - Complexity classes

Assigning Cost – Basic Operations

- Network
- Storage
- Operations
  - Construction
  - Successful verification
  - Failed verification
- Subversion
- Guessing
  - Various ways

- Supports very high precision
- Difficulty depends on precision
- Possibly subjective

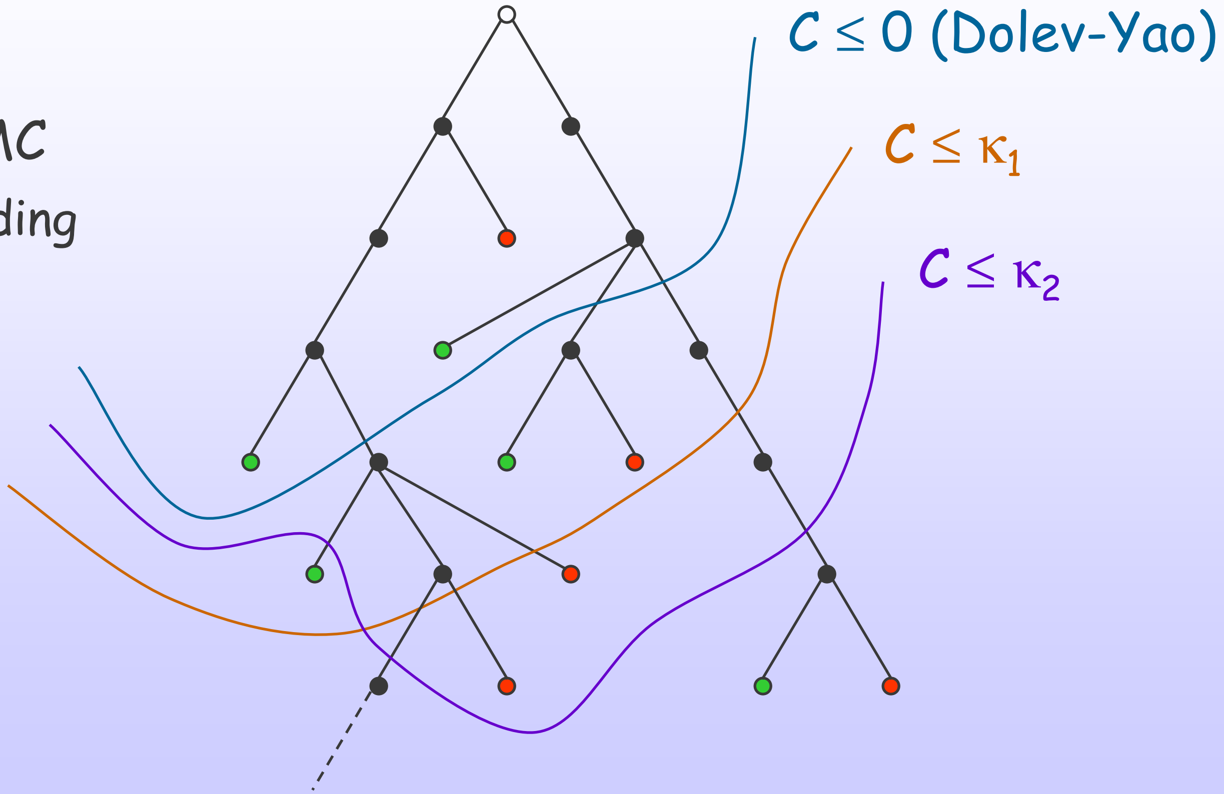
Assigning Costs – Traces & Scripts

- Traces: $\kappa(T)$
  - ➢ Add up basic costs
    - Monotonic costs: time, energy, …
    - Non-monotonic: space, …
- Scripts: $\kappa(S)$
  - ➢ Interval arithmetic
    - Script alternative

Quantitative Security Analysis

A model checking view

- Explicit state MC
  - Direct
- Symbolic MC
  - Via encoding



Threshold Analysis

- $\kappa(S_{NR}) \leq \kappa_{HW/HCI}$?
  - Cost of normal run acceptable?
    - PDAs, cell phones, …
- $\kappa(S_A) \leq \kappa_I$?
  - Cost of attack/defense acceptable?
  - Cost of candidate attack vs. resources
    - Non Dolev-Yao operations
- $\min x.\ \ \kappa(S_A(x)) \geq \kappa_{I++}$?
  - Design protocol
  - Fine-tuning parameters

Comparative Analysis

- $\kappa(S_{A1}) \leq \kappa(S_{A2})$?
  - Comparing attacks
    - Protocol can always be attacked
- $\kappa(S^{P1}) \leq \kappa(S^{P2})$?
  - Comparing protocols
- $\kappa^{B}(S_{A}) \leq \kappa^{I}(S_{A})$?
  - Comparing attack and defense costs
    - Denial of Service

Conclusions

- Quantitative protocol analysis
  - Cost conscious attacks (non Dolev-Yao)
  - Fine-Grained specification languages (MSR)
- Related work
  - C. Meadows: Cost framework for DoS
  - G. Lowe: guessing attacks
  - D. Tomioka, et al: cost for spi-calculus
- Future work
  - Attack costs: WEP
  - DoS aware protocols: JFK, client puzzles, bins
  - Protocol analysis as optimization problem
  - Economics of network security
  - Complexity-based costs and mixing probability

Typical Client/Server Exchange

Client			Server	
s^c_q	t^c_q	$\xrightarrow{\text{request}}$	s^s_q	t^s_q
s^c_c	t^c_c	$\xleftarrow{\text{challenge}}$	s^s_c	t^s_c
s^c_r	t^c_r	$\xrightarrow{\text{response}}$	$-(s^s_q+s^s_c)$	t^s_r
s^c_o	t^c_o	$\xleftarrow{\text{ok}}$	0	t^s_o

$\le T$

$\le B$

Time DoS

1. $\varepsilon \xrightarrow{\wp} t^s_q$

- Service rate: $1/t^s_q$
  - Usually dominated by networking costs

2. $t^c_q \xrightarrow{q} t^s_q$

   $0 \xleftarrow{c} t^s_c$

   $\dashrightarrow$

- Service rate
  - $1/(t^s_q + t^s_c)$
- Attack rate
  - $1/t^c_q$

3. $t^c_q \xrightarrow{q} t^s_q$

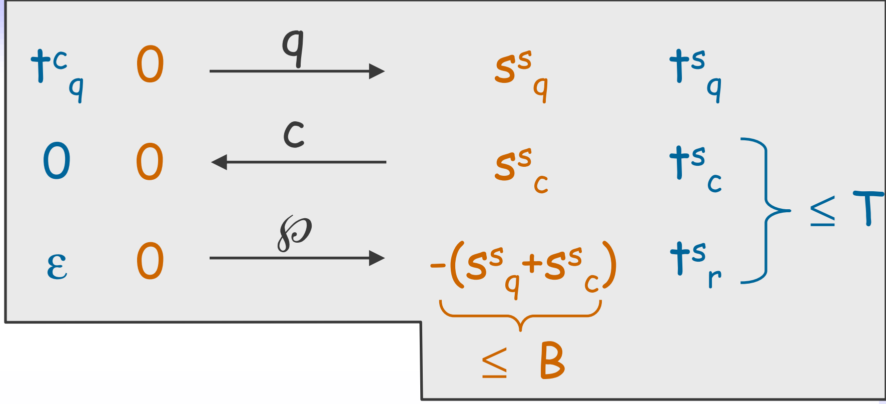
   $0 \xleftarrow{c} t^s_c$

   $\varepsilon \xrightarrow{\wp} t^s_r$

- Service rate
  - $1/(t^s_q + t^s_c + t^s_r)$
- Attack rate
  - $1/t^c_q$

Better attack

Space DDoS



- Max concurrent requests
  - $n(B) = B \,/\, (s^s_q + s^s_c)$
- Optimal time-out
  - $t_{min} < T$
  - $T \leq (t^s_q + t^s_c)^* \, (n(B) - 1)$
- Example
  - $s^s_q + s^s_c = 128$ b
  - $t^s_q + t^s_c = 100$ ms
  - $t_{min} = 90$ s
  - $n(B) = 10{,}000$

B = 1.28 Mb
T ≤ ~16 min