Authentication Tests and the Structure of Bundles

Joshua D. Guttman
F. Javier Thayer

September 2000

Today's Lecture

- Authentication Tests:
  - How to find out
    what a protocol achieves
  - How to prove it achieves that
  - Methods to establish
    - Secrecy (especially of keys)
    - Authentication
- Justifying authentication tests
  - Equivalence of bundles
  - Graph operations to simplify bundles
    - Well-behaved bundles
  - Paths through bundles
  - Transforming edges and pedigrees
  - The secrecy theorem
  - Authentication test theorems

Goals for this Hour

- Justify authentication test method
  - Use three ideas
    - Use equivalence relation on bundles
      Security goals invariant
      under equivalence
    - Focus on “well-behaved” bundles
      For every bundle, an equivalent
      well-behaved bundle exists
    - Consider paths through bundles
- Tomorrow: Apply same proof methods
  to protocol mixing

Definition: Bundles

A subgraph $\mathcal{C}$ of G_Σ is a *bundle* if $\mathcal{C}$ is finite and causally well-grounded, which means:

1. If $n_2 \in \mathcal{C}$ negative,
   there is a unique $n_1 \rightarrow n_2$ in $\mathcal{C}$
   (everything heard was said)
2. If $s \downarrow i + 1 \in \mathcal{C}$, then
   $s \downarrow i \Rightarrow s \downarrow i + 1$ in $\mathcal{C}$
   (everyone starts at the beginning)
3. $\mathcal{C}$ is acyclic
   (time never flows backward)

Causal partial ordering $n_1 \preceq_\mathcal{C} n_2$ means n_2 reachable from n_1 via arrows in $\mathcal{C}$

Induction: If $S \subset \mathcal{C}$ is a non-empty set of nodes, it contains $\preceq_\mathcal{C}$-minimal members

Equivalent Bundles

- Bundles $\mathcal{C}, \mathcal{C}'$ are equivalent iff they have the same regular nodes
  - Written $\mathcal{C} \equiv \mathcal{C}'$
  - Penetrator nodes may differ arbitrarily
  - Ordering $\preceq$ may differ arbitrarily
- Authentication goals invariant under equivalence
- Secrecy goals may be expressed in invariant form

  Define v "uncompromised" in $\mathcal{C}$ to mean:

  if for all $\mathcal{C}' \equiv \mathcal{C}$ and $n \in \mathcal{C}'$,

  then $v \not\sqsubseteq_\emptyset \mathrm{term}(n)$
- "Regular nodes" means non-penetrator nodes

  $v \sqsubseteq_\emptyset t$ concatenating v to other terms yields t

  (v is visible in t, not protected by encryption)

Paths

and

Normality

Graph Operations

- A graph operation may:
  - Delete penetrator strands
  - Add edges $n \to n'$
    with term$(n) = +a$, term$(n') = -a$
  - Delete edges $n \to n'$
- A graph operation yields graph $\mathcal{C}'$
  - $\mathcal{C}'$ not necessarily a bundle
  - But if it is a bundle, then
    $\mathcal{C}' \equiv \mathcal{C}$

Loneliness

- A lonely node in a graph has no edge
  - No incoming edge if negative
  - No outgoing edge if positive
- In definition of bundle:
  - Lonely negative nodes are ruled out:
    You can't hear something if nobody says it
  - Lonely positive nodes are allowed:
    Nobody hears what you say

Gregariousness

- A gregarious node in a graph has
  - Several incoming edges if negative
  - Several outgoing edges if positive
- In definition of bundle:
  - Gregarious negative nodes are ruled out:
    Hear the soloists, not the choir
  - Gregarious positive nodes are allowed:
    Many people hear your words

When are Graph Operations OK?

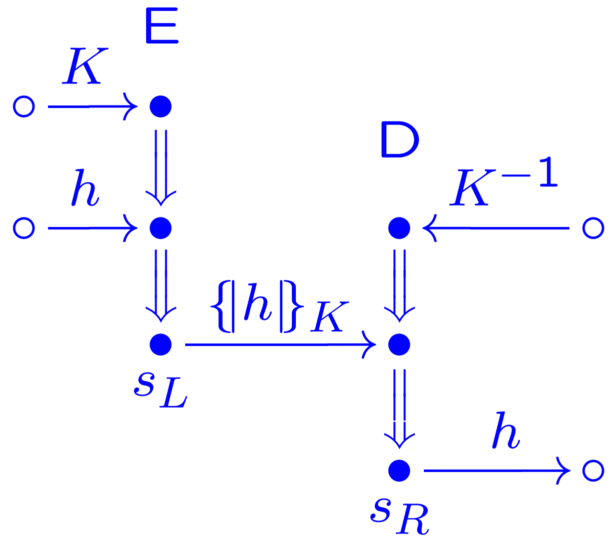
Suppose $\mathcal{C}'$ is obtained from bundle $\mathcal{C}$ by a graph operation such that

- For any edge new $n \mapsto n'$ of $\mathcal{C}'$, $\quad n \preceq_{\mathcal{C}} n'$
- $\mathcal{C}'$ has no lonely or gregarious negative nodes

Then

- $\mathcal{C}'$ is a bundle
- $\mathcal{C}' \equiv \mathcal{C}$
- The ordering $\preceq_{\mathcal{C}'}$ on $\mathcal{C}'$ weakens the ordering $\preceq_{\mathcal{C}}$ on $\mathcal{C}$

E-D Redundancies



C-S Redundancies

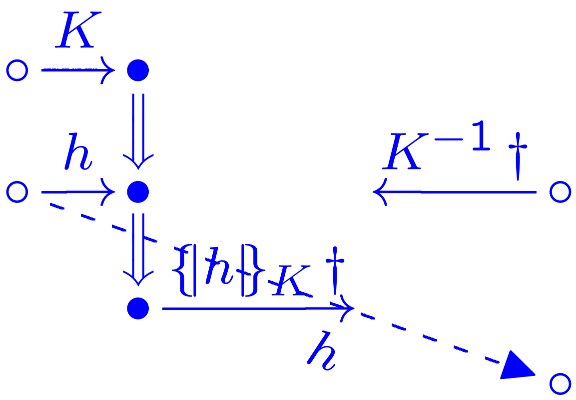
$$
\begin{tikzcd}
 & \mathsf{C} & & \\
\circ \arrow[r, "g"] & \bullet \arrow[d, Rightarrow] & & \\
\circ \arrow[r, "h"] & \bullet \arrow[d, Rightarrow] & \mathsf{S} & \\
 & \underset{s_L}{\bullet} \arrow[r, "gh"] & \bullet \arrow[d, Rightarrow] & \\
 & & \bullet \arrow[r, "g"] \arrow[d, Rightarrow] & \circ \\
 & & \underset{s_R}{\bullet} \arrow[r, "h"] & \circ
\end{tikzcd}
$$

Redundancy Elimination

- Any bundle $\mathcal{C}$ is equivalent to a bundle $\mathcal{C}'$ with no redundancies. Moreover,
  - Penetrator nodes of $\mathcal{C}'$ is a subset of penetrator nodes of $\mathcal{C}$
  - The ordering $\prec_{\mathcal{C}'}$ weakens the ordering $\prec_{\mathcal{C}}$
- Proof: Next two slides
- Consequence: Can assume attacker always

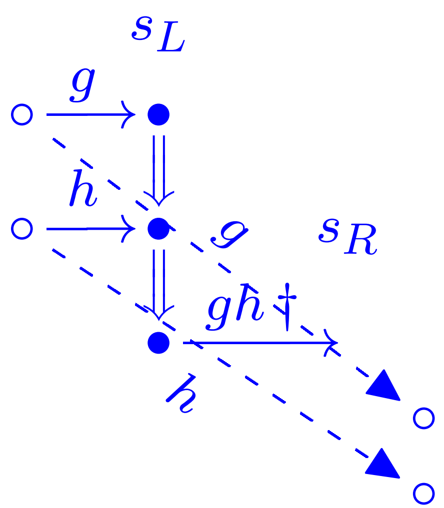
First	Takes things apart
Next	Puts things together
Then	Delivers results

E-D Redundancy Elimination



† Discarded message

C-S Redundancy Elimination

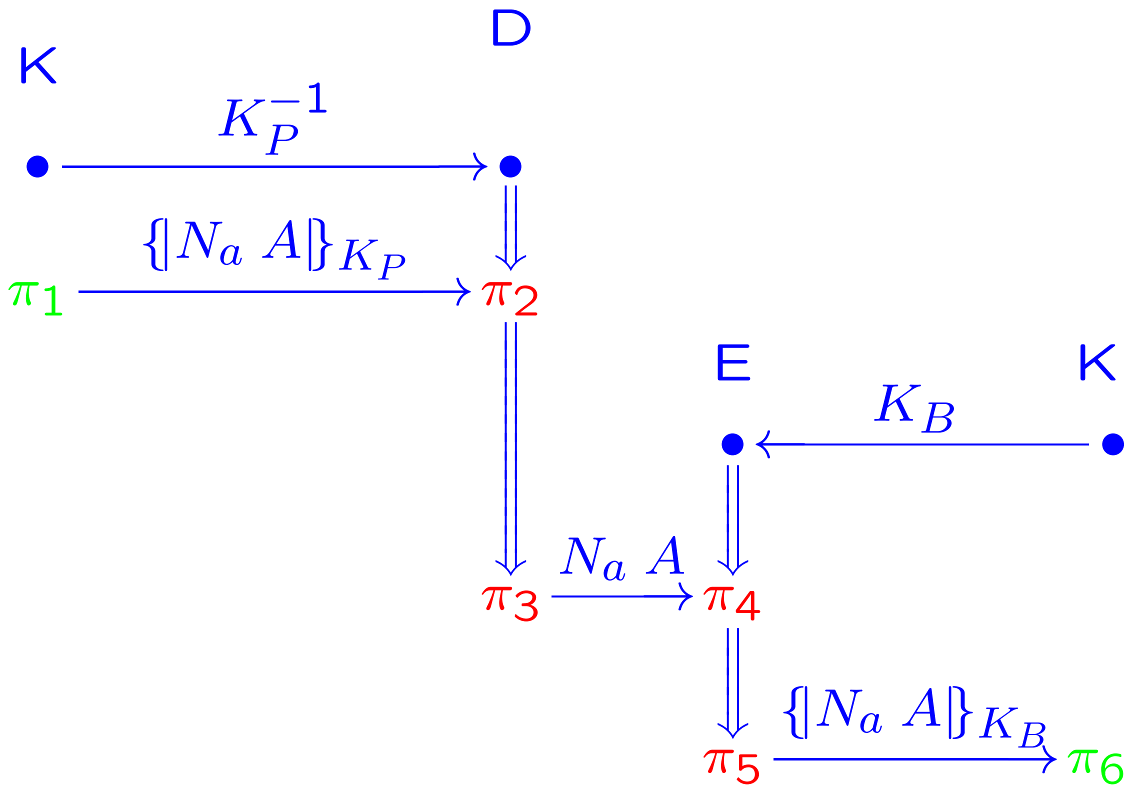


† Discarded message

Paths

- $m \Rightarrow^+ n$ means
  n occurs after m on the same strand
- $m \longmapsto n$ means either 1 or 2:
  1. $m \rightarrow n$
  2. $m \Rightarrow^+ n$ where
     term(m) negative and
     term(n) positive
- Path p through $\mathcal{C}$: sequence
  $p_1 \longmapsto p_2 \longmapsto \cdots \longmapsto p_k$
  - Typically assume p_1 positive node, p_k negative node
  - Notation: $|p| = k$, $\ell(p) = p_k$
- Penetrator path: p_j penetrator node,
  except possibly $j = 1$ or $j = k$

A Penetrator Path



Construction and Destruction

- A $\Rightarrow^+$-edge between penetrator nodes is
  - Constructive if part of a E or C strand
  - Destructive if part of a D or S strand
  - Initial if part of a K or M strand
- Constructive edge followed by a destructive edge
  Possible forms:
  - Node on $\mathsf{E}_{h,K}$ immediately followed by
    node on $\mathsf{D}_{h,K}$
    (for some h, K)
  - Node on $\mathsf{C}_{g,h}$ immediately followed by
    node on $\mathsf{S}_{g,h}$
    (for some g, h)
- This uses freeness of term algebra

Normality

- Bundle $\mathcal{C}$ normal iff

  No penetrator path p has constructive $\Rightarrow$ edge before destructive $\Rightarrow$ edge

- Any bundle is equivalent to a normal one
  - Eliminate redundancies
  - No other constructive/destructive pairs by freeness

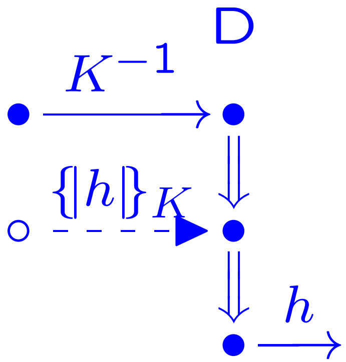
Rising and Falling Paths

- Definitions: (p a penetrator path)

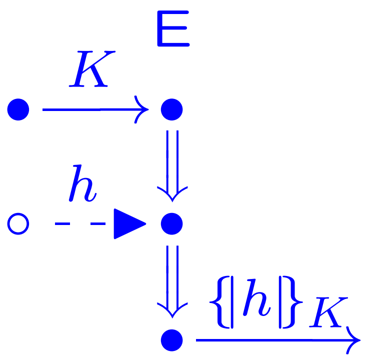
  Rising term$(p_i) \sqsubseteq$ term(p_{i+1})

  Falling term$(p_{i+1}) \sqsubseteq$ term(p_i)

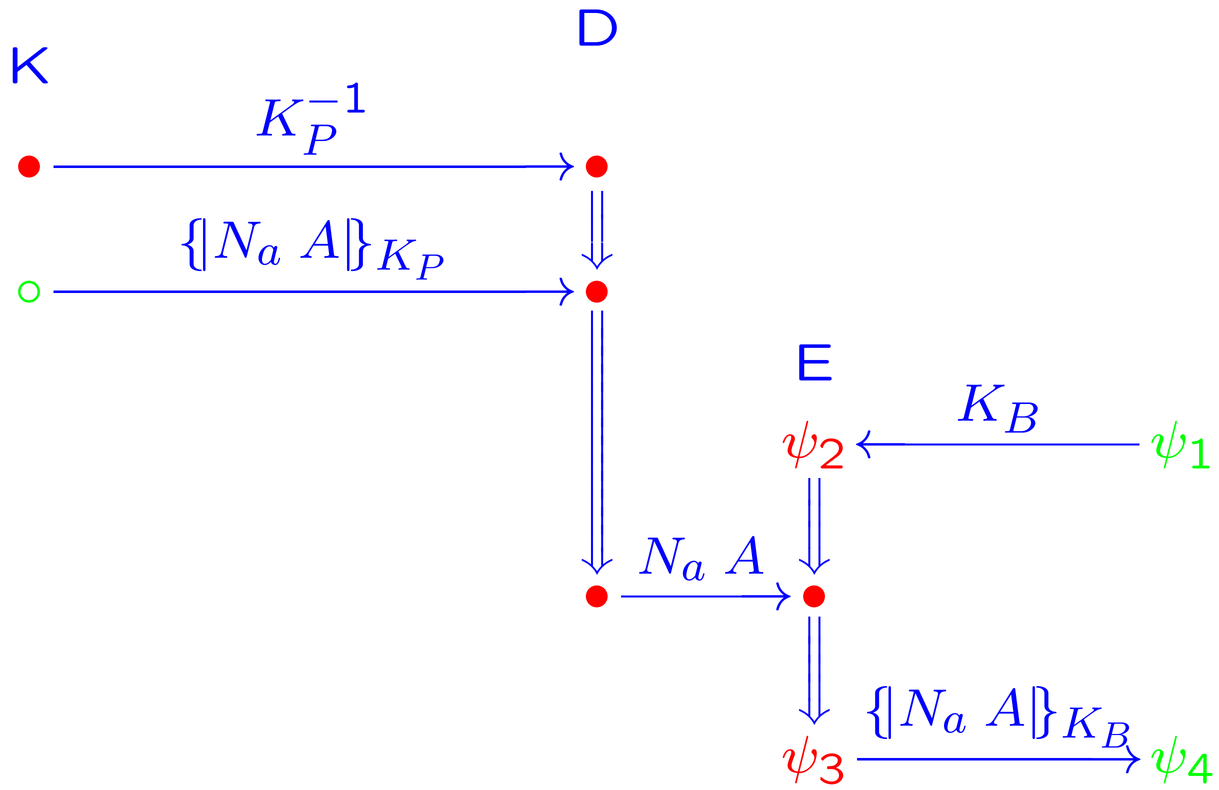
- Destructive paths may not be falling:



  Constructive paths may not be rising:



Another Penetrator Path



Paths that Avoid Key Edges

- If p is destructive and p never traverses D-key edge then p is falling

  $$\mathsf{term}(\ell(p)) \sqsubseteq \mathsf{term}(p_1)$$

- If p is constructive and p never traverses E-key edge then p is rising

  $$\mathsf{term}(p_1) \sqsubseteq \mathsf{term}(\ell(p))$$

- If bundle normal and p avoids key edges

  $p = q \ \to \ q'$
  q falling
  q' rising

- $\mathsf{term}(\ell(q)) = \mathsf{term}(q'_1) = \mathsf{pbt}(p)$
  called "path bridge term"

  $\mathsf{pbt}(p) \sqsubseteq p_1$
  $\mathsf{pbt}(p) \sqsubseteq \ell(p)$

Classifying Penetrator Paths

- Let p penetrator path; traverse backward.
  It may either:

	Reach an initial penetrator node (M, K)
or	Reach a non-initial E- or D-key edge
or	p_1 is regular

- If penetrator path p is useful, then either:

	$\ell(p)$ is regular
or	$\ell(p)$ is a key edge

- All penetrator activity divides into paths p
  where p never traverses key edge

$p_1, \ell(p)$ both regular	
p_1 initial, $\ell(p)$ reg.	* term(p_1) $\sqsubseteq$ term($\ell(p)$)
p_1 regular	$K =$ term($\ell(p)$)
p_1 a K-node	* $p = \quad p_1 \rightarrow p_2$

* If bundle $\mathcal{C}$ normal

Falling Penetrator Paths

- Suspect p_i negative with $1 < i < |p|$
  Then term(p_i) not atomic and

  either term$(p_i) = \{\!|h|\!\}_K$ and p_i on D
  or term$(p_i) = g\ h$ and p_i on S

- If p_i positive, term$(p_i) =$ term(p_{i+1})
- Suppose p traverses D with key edge K^{-1}
  only if $K \in \mathfrak{K}$

  Then term$(\ell(p)) \sqsubseteq_{\mathfrak{K}}$ term(p_1)

- Definition: $t_0 \sqsubseteq_{\mathfrak{K}} t$ iff
  t can be built from t_0 using only
  - concatenation (with anything)
  - encryption using $K \in \mathfrak{K}$

  $$\cdots \{\!|\cdots t_0 \cdots|\!\}_K \cdots$$

Well-Behaved

Bundles

Well-Behaved: Definition

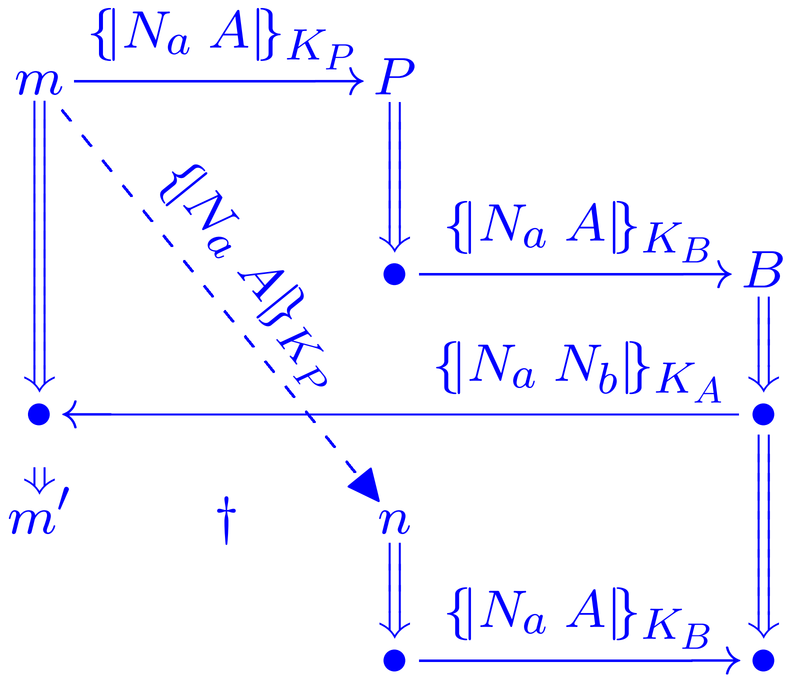
- A bundle is well-behaved if
  - Normal
  - Efficient
  - Has simple bridges
- Will define “efficient,” “simple bridges”
- Every bundle is equivalent to a well-behaved bundle

An Inefficient Bundle

$$
\begin{tikzcd}
m \arrow[r, "\{\!|N_a\ A|\!\}_{K_P}"] \arrow[dd, Rightarrow] & P \arrow[d, Rightarrow] & \\
 & \bullet \arrow[r, "\{\!|N_a\ A|\!\}_{K_B}"] & B \arrow[d, Rightarrow] \\
\bullet \arrow[d, Rightarrow] & & \bullet \arrow[ll, "\{\!|N_a\ N_b|\!\}_{K_A}"'] \arrow[dd, Rightarrow] \\
m' \arrow[r, "\dagger" near start, "\{\!|N_a\ A|\!\}_{K_P}"] & n \arrow[d, Rightarrow] & \\
 & \bullet \arrow[r, "\{\!|N_a\ A|\!\}_{K_B}"] & \bullet
\end{tikzcd}
$$

- Note: This protocol is fictitious!

An Efficient Bundle



Efficient Bundles

- In efficient bundle,
  penetrator avoids unnecessary regular nodes
- $\mathcal{C}$ is an efficient bundle iff:
  If m, n are nodes

  n negative penetrator node
  every component of n is a component of m

  Then there are no regular nodes m' such that

  $m \prec m' \prec n$
- For all $\mathcal{C}$, there exists $\mathcal{C}'$ where

  $\mathcal{C} \equiv \mathcal{C}'$
  $\mathcal{C}'$ efficient, normal

Simple Bridges

- Simple term is either

  An atomic value K, N_a, etc.
  An encryption $\{|h|\}_K$

  Anything but a concatenation
- $\mathcal{C}$ has simple bridges iff

  whenever p a penetrator path
  $\mathrm{pbt}(p)$ is simple

- Every $\mathcal{C}$ has an equivalent $\mathcal{C}'$ with simple bridges

$$
\begin{tikzcd}
 & \mathsf{S} & & \\
\circ \arrow[r, "gh"] & \bullet \arrow[d, Rightarrow] & \mathsf{C} & \\
 & \bullet \arrow[r, "g"] \arrow[d, Rightarrow] & \bullet \arrow[d, Rightarrow] & \\
 & \bullet \arrow[r, "h"] & \bullet \arrow[d, Rightarrow] & \\
 & & \bullet \arrow[r, "gh"] & \circ
\end{tikzcd}
$$