Cyberday 2015

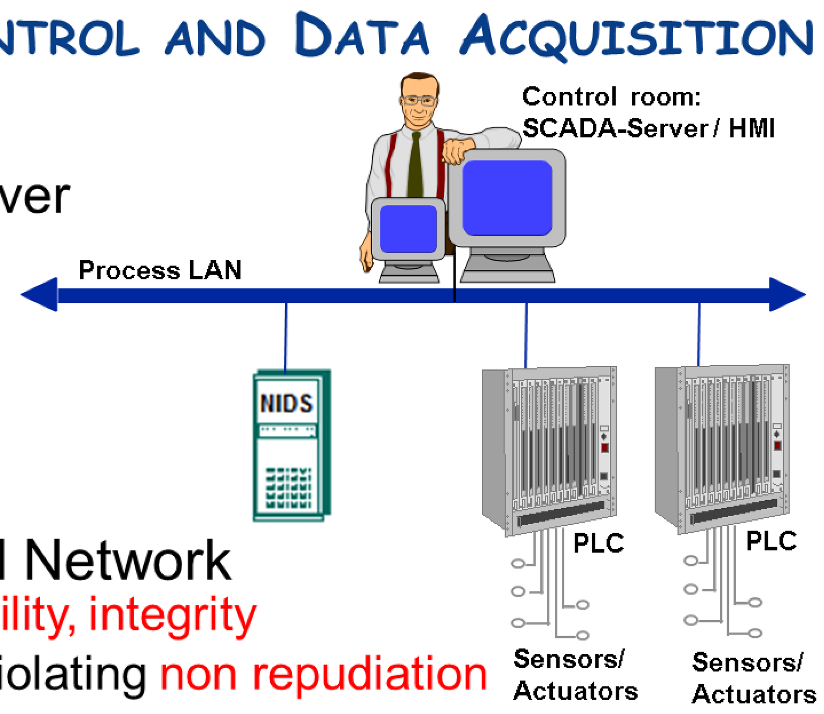
יום עיון בסייבר ואבטחת מידע

A Statechart-Based Anomaly Detection Model for Multiplexed SCADA Streams

Amit Kleinmann and Avishai Wool

SCADA Supervisory Control and Data Acquisition

- **Data Acquisition -** Sensors
- **Control -** RTU/PLC, MTU, Server
- **Network Communications**
  - Query-Response Protocol
- **Data Presentation –** HMI



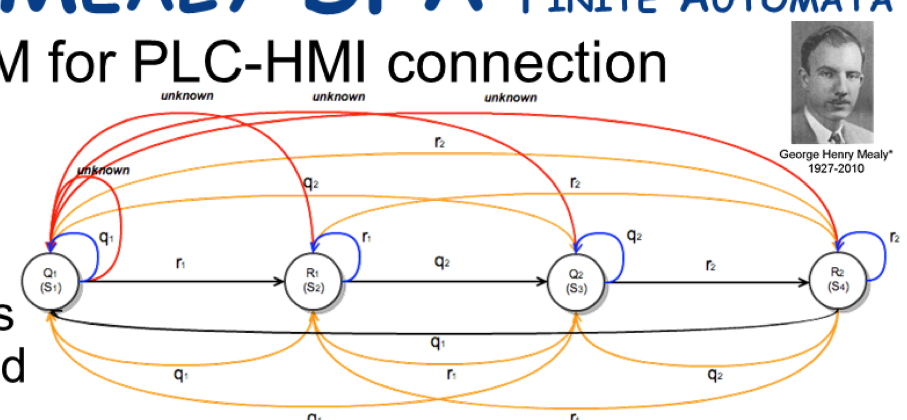
Threats

- Gaining access to the Control Network
  - Violating: confidentiality, availability, integrity
- Deny committing an attack - Violating non repudiation

The GW Model: Mealy DFA (Deterministic Finite Automata)

Learning phase: builds FSM for PLC-HMI connection
- State represents a valid Msg
- Symbol – PDU fields

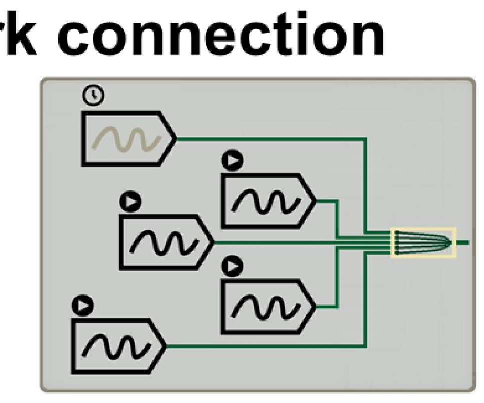
Enforcement phase
- Irregular query/response occurs => irregular behavior is detected



Multiplexed Cyclic Patterns

Multiple **streams share** the same **network connection**

- **Multi-Threaded HMI** - Each thread:
  - Has its own scan frequency
  - Independently scans a separate set of control registers
- **Push data**
  - The HMI subscribes to a register range
  - The PLC asynch. sends register values

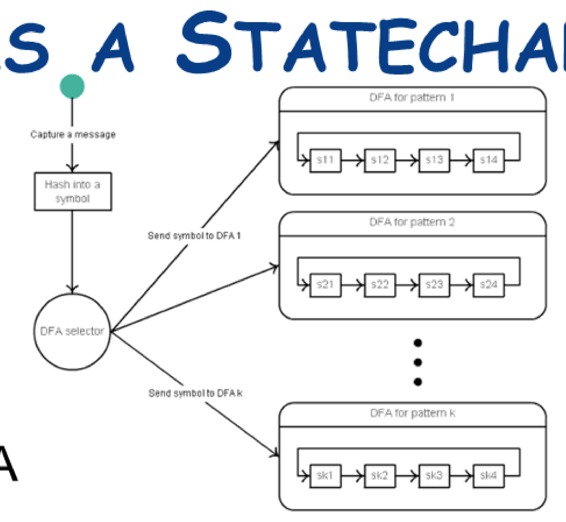


Modeling by a single DFA produces: Very large DFA and High false-alarm rate

Modeling the Traffic as a Statechart

Modeling each HMI-PLC channel as a separate Statechart with:

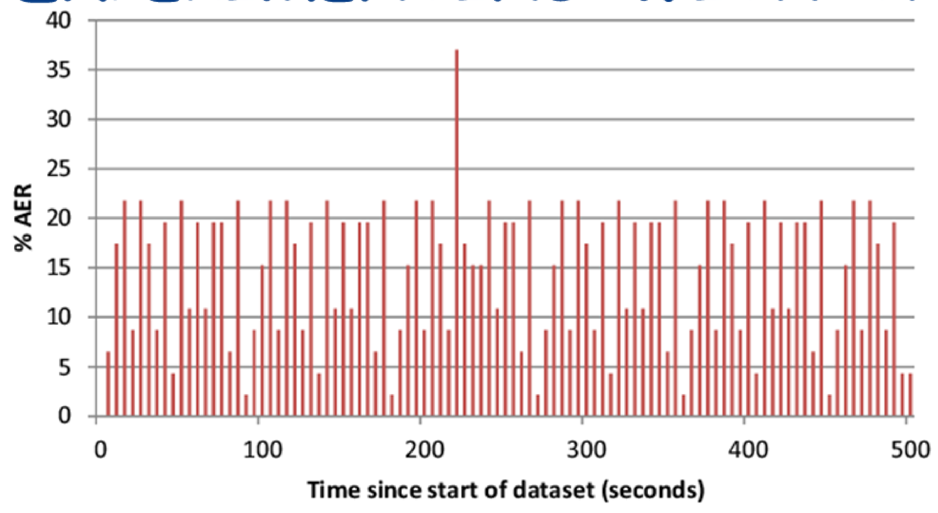
- Multiple DFAs, one per cyclic pattern, and
- A DFA-selector:
  - De-multiplexes incoming symbol stream
  - Sends each symbol to its respective DFA



The Statechart Learning Phase

1. Split the channel's input stream into multiple sub-channels.
2. ∀sub-channel create a DFA using the GW learning algorithm
   - During the DFA learning stage, for each state r in the DFA's pattern - calculate and keep the Time to Next State by $TNS(r)$:
     The average time difference between r and its immediate successor in the cyclic pattern (along the "Normal" transition).
3. Create the DFA-selector's mapping ϕ

Experimenting with the S7-0x72 Data



Results of applying the naive DFA model on a dataset of real Siemens S7-0x72 SCADA traffic

Dataset #	1	2
Duration	560 Sec.	2632 Sec.
TCP Packets	15875	67585
S7 Packets	4600	23553
AER	9.19	9.16

Dataset #	1		2	
DFA type	Naiv	Schrt	Naiv	Schrt
Model size	62	3	12	3
False alrm %	14.54	0.11	12.98	0

Results of applying both models on Siemens S7-0x72 SCADA traffic

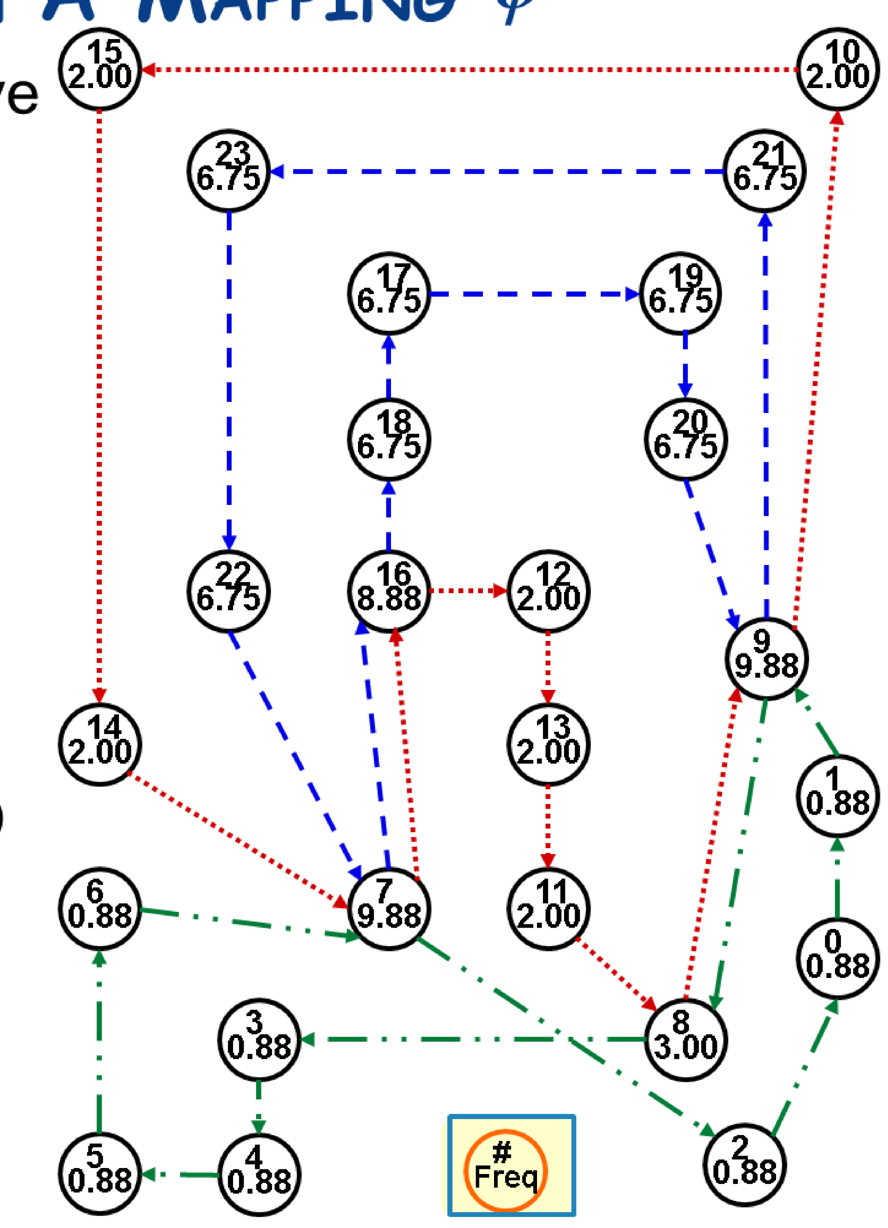
Symbol-to-DFA Mapping ϕ

$\phi(s)$ ≡ the set of DFAs that have symbol s in their pattern

Different cases:

1. Each sub-channel has a unique set of symbols.
   - $\phi(s)$ = {D} => s is sent to D
2. The patterns overlap
   - Some symbols belong to multiple sub-channels
   - $|\phi(s)| > 1$ => the selected DFA D is the member of $\phi(s)$ for which - the absolute diff. between:
     - the current time (during enforcement) and
     - the predicted arrival time $T_{pred}(s;D)$

     is minimal



Calculating the Predicted Arrival Time

During the enforcement stage - each DFA D retains:

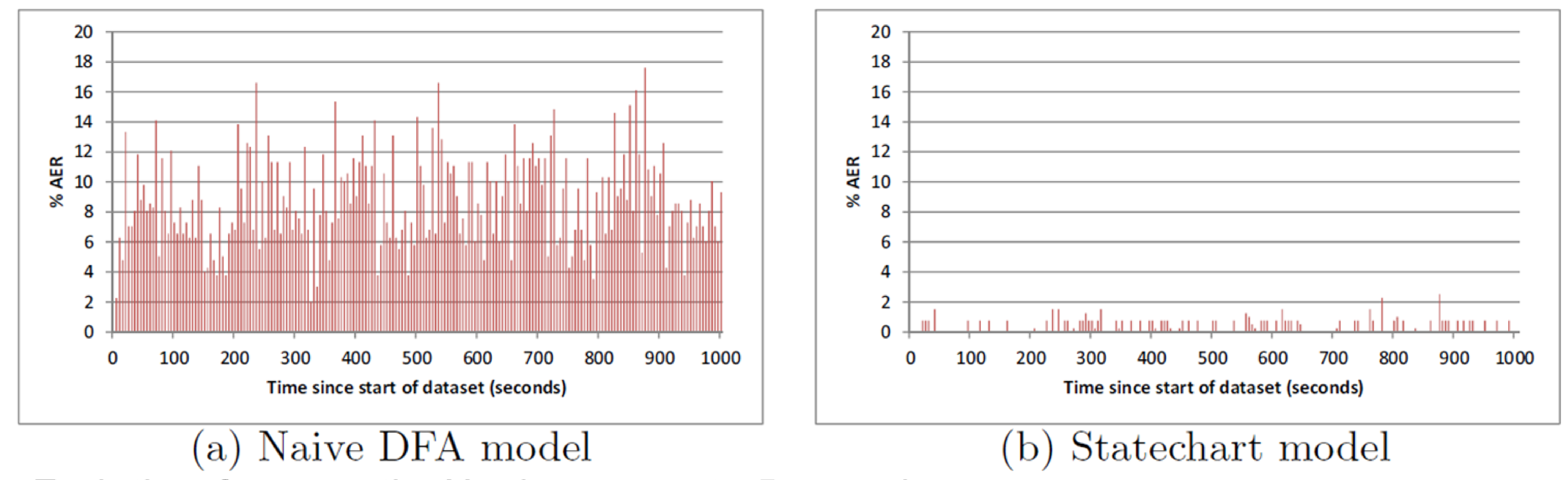
- The identifier of its last state
- $T_{last}(D)$ ≡ the time-stamp of the last symbol it processed

$T_{pred}(s;D)$ of a symbol s for a DFA $D \in \phi(s)$ which is at state q is calculated as follows:

1. Identify the tentative state q' that DFA D transitions to from state q upon symbol s.
2. $P(q; q')$ ≡ the path of DFA states from q to q' along the "Normal" transitions (not including q').

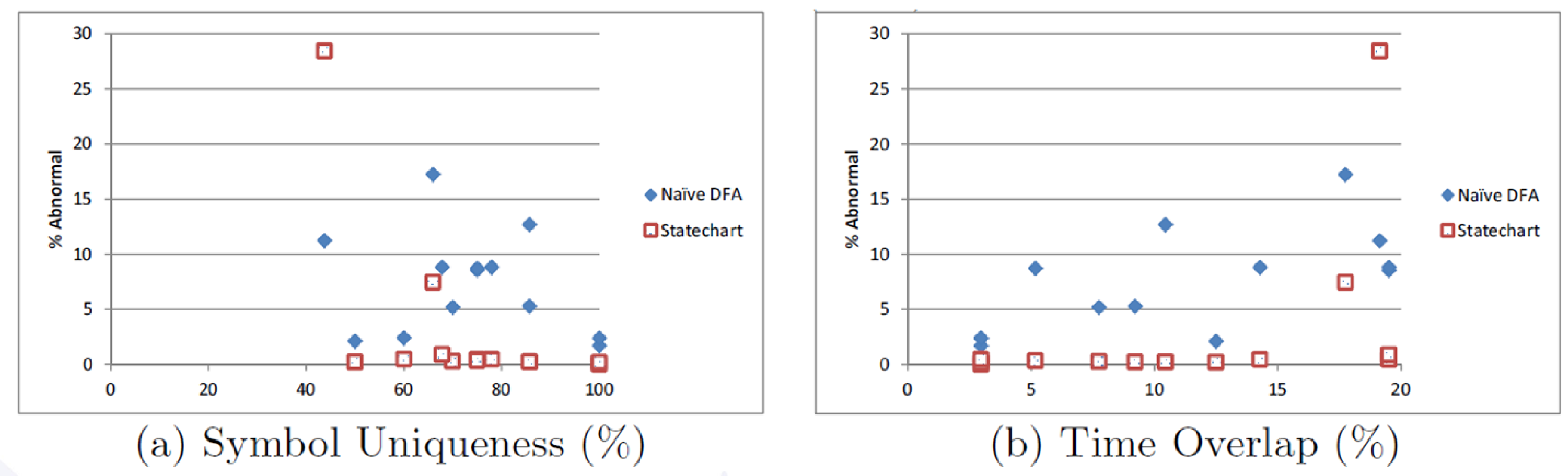
$$T_{pred}(s;D) = T_{last}(D) + \sum_{r \in P(q;q')} TNS(r)$$

False-Alarm Rate on Synthetic Dataset



(a) Naive DFA model (b) Statechart model

Each time frame on the X axis represents 5 seconds.
The Y axis shows the false alarm frequency as % of the AER for each time period.



(a) Symbol Uniqueness (%) (b) Time Overlap (%)

The false alarm rates as a function of the Symbol Uniqueness and Time Overlap

TCE technion computer engineering center